

# **Rigorous Numerical Enclosures for Control Affine Problems**

Dissertation  
zur Erlangung des Doktorgrades  
der Mathematisch-Naturwissenschaftlichen Fakultät  
der Universität Augsburg

vorgelegt von  
**Albert Marquardt**

Augsburg, November 2005

Erstgutachter: Prof. Dr. Fritz Colonius

Zweitgutachter: Prof. Dr. Kunibert G. Siebert

Tag der mündlichen Prüfung: 10. Januar 2006

# Contents

<b>Contents</b>	<b>iii</b>
<b>Introduction</b>	<b>1</b>
<b>1 Problem Formulation and the Toolbox</b>	<b>5</b>
1.1 Control Affine Systems . . . . .	6
1.2 Interval Arithmetics . . . . .	9
1.3 Multi-Index Theory . . . . .	14
1.3.1 Definition . . . . .	15
1.3.2 Shuffle-Product . . . . .	17
1.3.3 Combinatorial Selection . . . . .	18
1.3.4 Insertion Operator . . . . .	21
1.3.5 Serial Number Representation . . . . .	24
<b>2 Fliess-Expansions</b>	<b>29</b>
2.1 Lie derivatives . . . . .	30
2.2 Iterated Integrals . . . . .	31
2.3 Integral representations . . . . .	35
2.4 Fliess-expansions . . . . .	41
2.5 Uniqueness of Fliess-expansions . . . . .	43
2.6 Operations on Fliess-expansions . . . . .	52
2.6.1 Special Fliess-expansions . . . . .	52
2.6.2 Addition . . . . .	53
2.6.3 Multiplication . . . . .	53
2.6.4 Taylor-expansions of Fliess-expansions . . . . .	64
2.6.5 Division . . . . .	66

<b>3</b>	<b>Numerical Algorithm</b>	<b>69</b>
3.1	Coefficients for the Solution . . . . .	69
3.2	Enclosure for the Remainder . . . . .	73
3.3	Initial Enclosure . . . . .	77
3.4	Error Propagation . . . . .	78
3.5	The Algorithm . . . . .	81
<b>4</b>	<b>Application</b>	<b>87</b>
	<b>Bibliography</b>	<b>93</b>

# Introduction

The aim of this thesis is the formulation of a numerical algorithm for finding guaranteed bounds for all solutions of a nonlinear control affine system with bounded control functions and an initial interval on a given time range. The main tools are the theory for Fliess-expansions, the automatic differentiating method and interval arithmetics.

The motivation comes originally from the numerical computation of viability kernels (see Aubin [1]), reachability sets and control sets (see Colonius, Kliemann [5]) with set valued numerics, in particular with the program package GAIO (see Dellnitz, Froyland, Junge [6]). GAIO is made for the analysis of ordinary differential equations and difference equations on bounded state spaces with subdivision techniques. The extension of the subdivision algorithms on control systems was done by Szolnoki [32] in 2001. He computed many two- and three-dimensional examples for control systems with GAIO. Grüne [11] introduced an adaptive subdivision algorithm to find the boundaries of reachability sets. Marquardt [28] computed a time periodic oscillator equation and Gayer [10] analysed the bifurcation of control sets for perturbed systems.

The idea of subdivision algorithms is, starting with an initial collection of sets, to subdivide each set into two parts and select from the newly generated finer collection those which satisfy a selection criterion. This criterion is based on the reachability relations amongst the sets in the given collection for a small time step. For the reachability information of one set the initial value problem has to be solved for each point, where all allowed control functions are taken into account. This is realized by choosing a sufficient number of test points in the appropriate set and simulate the differential equation numerically for different constant control functions with standard ODE solvers. The crucial part is to determine a chosen number of test points and control function as sufficient. In practice collections consist of boxes or in other words full dimensional intervals. This leads us directly to the wide field of interval analysis. In chapter 4 we will compute a simple example

with GAIO but with our new algorithm from interval analysis.

Starting point is an algorithm developed by Lohner [26] in 1988 to find the solution for an initial value problem by approximating it with a Taylor-expansion. This method was already known in numerics as the power series method. But therefore the derivatives up to a given order of the right hand side need to be computed. Lohner used a method which was known from automatic differentiating (see Rall [30]). There the right hand side was computed out of the power series for the solution. Therefore many mathematical operations, like the basic arithmetic operations and intrinsic functions, were defined for power series. So the series representations for the right and left hand side of the ODE could be compared coefficient-wise. This results in recursive equations for the coefficients of the power series.

Lohner gave a strategy to enclose the Lagrange remainder term of the Taylor-expansion on a given bounded time interval. Therefore an initial enclosure for all solutions is needed and Lohner developed a sufficient condition to identify an initial enclosure. After realizing this computations with interval arithmetics, the enclosure becomes guaranteed. Lohner extended the algorithm on initial interval problems by interpreting the initial interval as an error bound for the initial value. Then he enclosed the error propagation for the coefficients and the remainder of the solution's power series.

For control affine systems we proceed like Lohner for ODEs. Instead of Taylor-expansions the Fliess-expansions have to be used. The Taylor-expansions have a linear sequence of coefficients. The coefficients of Fliess-expansions have a tree structure and are indexed by multi-indices. In literature they are also known as the Chen-Fliess-expansions and conditions for their convergence are already given (cp. Isidori [17, chapter 3]). Grüne and Kloeden [12] used them to formulate numerical schemes to approximate the solution of control affine systems.

For computing the coefficients for the solution's Fliess-expansion using automatic differentiating the expansion has to be compared with the expansion for the integral of the systems' right hand side coefficient-wise. This can only be done if for an arbitrary function, which possesses a finite Fliess-expansion, every coefficient is unique. Fliess [9] gave already an algebraic motivated proof for the uniqueness of infinite expansions for solutions of control affine systems. In chapter 2 we will define finite Fliess-expansions formally and prove uniqueness of their coefficients in notions of analysis.

In order to get Fliess-expansions for the right hand side vector fields of the control affine system, we must be able to calculate with Fliess-expansions. First of all the basic arithmetic operations must be defined.

Where the addition is achieved coefficient-wise, the multiplication is unlike more difficult. It takes some effort to show that the product function of two Fliess-expansions is again a Fliess-expansion and to give a recursion formula for its coefficients. With the multiplication formula recursions for division and for arbitrary Taylor-expansions with Fliess-expansions in the argument can be derived.

Before we can formulate an appropriate algorithm we need to provide a little toolbox in chapter 1. After the definition of control affine problems and their solutions we will give some basic definitions from interval arithmetics following the book of Jaulin et al. [18]. We will illustrate the wrapping effect, which is a frequent source for overestimating results in interval analysis. We will show how to reduce it using the patching strategy, which is one out of a huge number of proposals to reduce the wrapping effect.

Secondly we will develop a small multi-index theory. Multi-indices are usually used to abbreviate the notation of higher order partial derivatives. Here we use an similar definition to denote higher order Lie-derivatives and we will determine iterated integrals with multi-indices. Therefore we will define multi-indices and operations on multi-indices in chapter 1. This helps us to perform arithmetics for the operators on the index-level. Apart from the shuffle-product, which was defined by Fliess [9], we define some very special operations like the selection and the insertion operator.

In chapter 3 we need to restrict the range of the control functions. This assumption is very natural, because even in the case of controllable linear systems any point can be reached in any time for unbounded control functions.

We will formulate the recursion formulas for the solution and for the remainder's enclosure. For the latter we need an initial enclosure of all solutions on the given time interval. We will formulate a criterion to identify an enclosure and a strategy to find it. Next we linearise the solution's coefficients in the initial value. This leads us to the propagation of the initial interval. At the end of chapter 3 we formulate and discuss the algorithm.

As a further important motivation for using interval analysis and verified numerics the field of numerical proofs should be mentioned. For instance Kapela and Zgliczynski [21] proved the existence of special periodic solutions for the  $N$ -body problem. Their idea is as follows. They assume some properties for a solution and formulate the so called Krawczyk operator. The Krawczyk operator needs an enclosure for all solution of an initial interval problem. Therefore the Lohner-algorithm [26] is used. With interval analysis it can be shown that the Krawczyk operator has a fixed point

a solution with the asserted properties. The computations are done with interval arithmetics which take into account the rounding and truncation errors. The fixed point equation is solved by the interval Newton method (see Krawczyk [24]) for finding enclosures for all roots of an algebraic equation. In Kapela [20] many very interesting types of solutions for the  $N$ -body problem are given and their existence is proved numerically. An analytic existence proof is for most of them impossible. Colonius and Kapela [3] proved already the existence of periodic and homoclinic orbits for several control systems. With theoretical knowledge for choosing the control functions they were able to reduce the control systems to ODEs.

Now I like to express my deep thanks and gratitude to Prof. Dr. Fritz Colonius for his instructive supervision, guidance and valuable advice expending much time and effort through the progress of this thesis. I thank my colleague Torben Stender for proofreading this thesis. Him and Tobias Gayer I am grateful for the pleasant time at the joint office. At the end I like to mention the “Graduiertenkolleg: Nichtlineare Probleme in Analysis, Geometrie und Physik” at the University of Augsburg. It made this work possible by a scholarship.

Augsburg, November 2005



# Chapter 1

## Problem Formulation and the Toolbox

In this chapter we first introduce the notion of control affine systems. The right hand side vector fields are numbered from 1 to  $m$ , which denotes the dimension of controls. In literature the uncontrolled part is indexed with 0, but we will need the 0-index for the time dependency. We will introduce a notion for the solution and prove its existence and uniqueness under the assumption of a Lipschitz condition on the right hand side vector fields.

Next we will introduce the interval arithmetic with the definitions we use in this thesis and we will have a look on the wrapping effect, which was already described by Lohner [27]. It often appears in numerical computations in more than one dimensions with interval arithmetics. The patching method is one out of many strategies to reduce this effect. We will use this method for the application in chapter 4.

Third in this chapter we will develop a multi-index theory. We will define some operations on the multi-index set like resorting, concatenation, separation and the shuffle-product. At the end we will identify the multi-index set with the nonnegative integer numbers and define the operators on  $\mathbb{N}_0$ , which are important for the implementation. The definition of multi-indices is recursive like the operators they will index in section 2. For this reason most of the proofs will be done by induction.

The interval arithmetic and the multi-index theory compose the toolbox we need to define and to analyse Fliess-expansions in the following two chapters.

## 1.1 Control Affine Systems

We define now the mathematical problem for which we will develop a numerical algorithm to enclose its solution. Many systems in control theory are control affine systems. These are differential equations where the right hand side vector fields are multiplied by scalar control functions.

**Definition 1.1 (control affine systems)**

$$\begin{aligned} \dot{x}(t) &= \sum_{\alpha=1}^m f^{\alpha}(t, x(t)) u^{\alpha}(t), \quad x : \mathbb{R} \rightarrow \mathbb{R}^n, \\ f^{\alpha} &: [\tau, T] \times \mathbb{R}^n \longrightarrow \mathbb{R}^n, \\ u^{\alpha} &\in \mathcal{U}, \quad \alpha \in \{1, \dots, m\}. \end{aligned} \tag{1.1}$$

Thereby  $n \in \mathbb{N}$  denotes the dimension of the state space and  $m \in \mathbb{N}$  is the dimension of control or number of scalar control functions. We assume the system to exist on a compact real time interval  $[\tau, T]$ . The space of control functions is the space of all integrable functions on the given time interval:

$$\mathcal{U}^m := \left\{ u : [\tau, T] \longrightarrow \mathbb{R}^n, \quad \begin{array}{l} u \text{ is Lebesgue-integrable} \\ \text{and essentially bounded.} \end{array} \right\}.$$

We call the vector of control functions  $u = (u^1, \dots, u^m)$  the control vector. In practice the first control function is defined by  $u^1 \equiv 1$ . Then by choosing  $u^{\alpha} \equiv 0$ , for  $\alpha = 2, \dots, m$  the system results in the corresponding uncontrolled problem:  $\dot{x}(t) = f^1(t, x(t))$ . Usually the vector fields  $f^{\alpha}$  are well known and smooth enough, i.e. all partial derivatives in  $t$  and  $x$  we need do exist. The control functions can be generated by a stochastic process or are just uncertain functions. Next we define the general solution of the control affine system.

**Definition 1.2 (general solution)** *The general solution of system 1.1 is denoted by*

$$\lambda : \mathbb{R} \times \mathbb{R} \times \mathbb{R}^n \times \mathcal{U}^m \longrightarrow \mathbb{R}^n.$$

*It depends on the initial time  $\tau$ , the initial value  $x(\tau) = x_0 \in \mathbb{R}^n$  and on the control vector  $u \in \mathcal{U}^m$ . The general solution is required to be differentiable in the first component and to satisfy the solution's identity*

$$\dot{\lambda}(t, \tau, x_0, u) = \sum_{\alpha=1}^m f^\alpha(t, \lambda(t, \tau, x_0, u)) u^\alpha(t) \quad (1.2)$$

*and the initial condition  $\lambda(\tau, \tau, x_0, u) = x_0$ , for all  $x_0 \in \mathbb{R}^n$  and all  $u \in \mathcal{U}^m$ .*

In some cases, mainly in section 2.3, when we deal with any arbitrary solution of the control system we will denote it with the state space variable  $x$ . We will use the denotation  $\lambda$  if the dependence on the initial conditions and the vector of control functions is relevant.

As for ordinary differential equations a sufficient criterion for the existence and uniqueness of the solution is the right hand side satisfying a Lipschitz condition in the second argument. We generalise the strategy of Walter [34] for control affine systems.

We assume the vector fields  $f^\alpha$  to fulfil a Lipschitz-condition in the second argument for all  $\alpha \in \{1, \dots, m\}$ , that is there exist positive real numbers  $0 < l^\alpha \in \mathbb{R}$  with

$$\|f^\alpha(s, y(s)) - f^\alpha(s, z(s))\|_\infty \leq l^\alpha \|y(s) - z(s)\|_\infty,$$

for any  $s \in [\tau, T]$  and  $y, z \in C([\tau, T], \mathbb{R}^n)$ , where  $\|\cdot\|_\infty$  denotes the maximum norm in  $\mathbb{R}^n$ . With the norm

$$\|y\| := \max_{s \in [\tau, T]} e^{-\mu(s-\tau)} \|y(s)\|_\infty,$$

for an arbitrary  $\mu > 0$  the function space  $C([\tau, T], \mathbb{R}^n)$  turns into a Banach-space. For given initial conditions  $\tau, x_0$  and control functions  $(u^1, \dots, u^m)$  we define the operator  $W : C([\tau, T], \mathbb{R}^n) \rightarrow C([\tau, T], \mathbb{R}^n)$

$$(Wy)(t) := x_0 + \sum_{\alpha=1}^m \int_{\tau}^t f^\alpha(s, y(s)) u^\alpha(s) ds.$$

In the next theorem we will prove the existence and uniqueness of a solution for the control system. Therefore we use the operator  $W$ , which will turn out to be a contraction on the Banach space.

**Theorem 1.3 (uniqueness of the solution)** *On condition that the right hand sides  $f^\alpha$ ,  $\alpha \in \{1, \dots, m\}$ , satisfy Lipschitz-conditions with Lipschitz-constants  $l^\alpha > 0$  on the given time interval  $[\tau, T]$  the solution  $\lambda(\cdot, \tau, x_0, u)$  of system 1.1 exists and is unique.*

**Proof:** For two functions  $y, z \in C([\tau, T], \mathbb{R}^n)$  we estimate the maximum norm from above at a fixed time  $t \in [\tau, T]$ :

$$\begin{aligned}
& \| (Wy - Wz)(t) \|_\infty \leq \\
& \leq \sum_{\alpha=1}^m \int_{\tau}^t \| (f^\alpha(s, y(s)) - f^\alpha(s, z(s))) u^\alpha(s) \|_\infty \, ds \\
& \leq \sum_{\alpha=1}^m \int_{\tau}^t l^\alpha \| (y(s) - z(s)) \|_\infty |u^\alpha(s)| e^{-\mu(s-\tau)} e^{\mu(s-\tau)} \, ds \\
& \leq \sum_{\alpha=1}^m l^\alpha \|y - z\| \operatorname{ess\,sup}_{s \in [\tau, T]} |u^\alpha(s)| \int_{\tau}^t e^{\mu(s-\tau)} \, ds \\
& \leq \frac{e^{\mu(t-\tau)}}{\mu} \|y - z\| \sum_{\alpha=1}^m l^\alpha \operatorname{ess\,sup}_{s \in [\tau, T]} |u^\alpha(s)|.
\end{aligned}$$

We choose  $\mu := 2 \sum_{\alpha=1}^m l^\alpha \operatorname{ess\,sup}_{s \in [\tau, T]} |u^\alpha(s)|$  and get immediately

$$\|Wy - Wz\| \leq \frac{1}{2} \|y - z\|.$$

Thus the operator  $W$  is a contraction on  $C([\tau, T], \mathbb{R}^n)$ . We can apply the Banach fixed point theorem and get a function  $x \in C([\tau, T], \mathbb{R}^n)$  as the unique solution of the fixed point equation

$$Wx = x.$$

This equation is the integral formulation of the control system 1.1. So every solution of the fixed point equation is a solution of the control system and vice versa. ■

We assume the vector fields  $f^\alpha, \alpha \in \{1, \dots, m\}$  to fulfil the Lipschitz condition. So the unique solution  $\lambda(t, \tau, x_0, u)$  is well-defined for all times  $t \in [\tau, T]$ , for all initial values  $x_0 \in \mathbb{R}^n$  and all control vectors  $u \in \mathcal{U}^m$  and solves the equation 1.1. For the proof of theorem 1.3 we used the version of the Banach fixed point theorem, which acts on a Banach space. The other version, which acts on a closed subset of a Banach space, we will use later in section 3.3.

## 1.2 Interval Arithmetics

In this section we give a short and of course non complete introduction into the wide field of interval arithmetics. In the 1950's several people introduced the idea of computing with error box around specific values. The field of interval arithmetic began in 1966 with a book by Moore [29]. Afterwards many articles and books were published till today on this subject. The first purpose of interval arithmetic was to handle rounding and truncation errors of numerical computations, which are caused by technical reasons.

Later the idea of enclosing the solution of problems from analysis was coming up. For instance the evaluation of a function was calculated via its Taylor series and the rigorous enclosure of the remainder term. So the wide field of interval analysis was born to handle discretisation errors and method errors.

We give a short introduction into the objects of interval arithmetics, we are using in this thesis. The definitions and notations are mainly taken from Jaulin, Kieffer, Didrit, Walter [18]. First we define intervals and interval vectors. We confine ourselves to closed intervals. So a one-dimensional interval is a closed and connected subset of  $\mathbb{R}$ . Higher dimensional intervals are rectangles, parallel to the coordinate axes.

**Definition 1.4 (intervals)** *The set of closed real intervals in one and in  $n \in \mathbb{N}$  dimensions is defined by:*

$$\mathbb{I} := \{[a, b] \mid a \leq b, a, b \in \mathbb{R}\},$$

$$\mathbb{I}^n := \left( \begin{array}{c} [v_1] \\ \vdots \\ [v_n] \end{array} \right), \text{ where } [v_1], \dots, [v_n] \in \mathbb{I}.$$

We want to calculate with intervals like we are used to do it with real numbers. Therefore we need to define operations on intervals.

**Definition 1.5 (operations on intervals)** *Let  $[a] \in \mathbb{I}^{n_1}$  and  $[b] \in \mathbb{I}^{n_2}$  be intervals with  $n_1, n_2 \in \mathbb{N}$  and let the operation  $a \circ b$  be defined for all  $a \in [a]$  and  $b \in [b]$ . Then we define with*

$$[a] \circ [b] := \{a \circ b \mid a \in [a], b \in [b]\}$$

the interval operation  $\circ$ . With

$$[[a] \circ [b]] := [\min [a] \circ [b], \max [a] \circ [b]]$$

we denote a interval enclosure  $[a] \circ [b]$ . The min- and max-functions are defined componentwise.

Note, that  $[[a] \circ [b]]$  exists only if the set  $[a] \circ [b]$  is bounded. One can identify easily the enclosures for the componentwise basic operations for two intervals  $[a], [b] \in \mathbb{I}^n$ ,  $n \in \mathbb{N}$ . We define  $\underline{a} := \min [a]$ ,  $\bar{a} := \max [a]$  and  $\underline{b}, \bar{b}$  respectively. Then we have

$$\begin{aligned} [a] + [b] &= [\underline{a} + \underline{b}, \bar{a} + \bar{b}], \\ [a] - [b] &= [\underline{a} - \bar{b}, \bar{a} - \underline{b}], \\ [a] * [b] &= [\min \{\underline{a}\underline{b}, \bar{a}\underline{b}, \underline{a}\bar{b}, \bar{a}\bar{b}\}, \max \{\underline{a}\underline{b}, \bar{a}\underline{b}, \underline{a}\bar{b}, \bar{a}\bar{b}\}] \end{aligned}$$

and with  $0 \notin [b]$

$$[a] / [b] = [\min \{\underline{a}/\underline{b}, \bar{a}/\underline{b}, \underline{a}/\bar{b}, \bar{a}/\bar{b}\}, \max \{\underline{a}/\underline{b}, \bar{a}/\underline{b}, \underline{a}/\bar{b}, \bar{a}/\bar{b}\}].$$

For the basic operations we discover the distributivity law not being valid any more. We have to replace it by the sub-distributivity.

**Lemma 1.6 (sub-distributivity)** *Let  $[a], [b], [c] \in \mathbb{I}$  be intervals. Then the following inclusion holds:*

$$[a]([b] + [c]) \subseteq [a][b] + [a][c].$$

**Proof:** By giving the definition of both sides, we easily see the correctness of the inclusion.

$$\begin{aligned} [a]([b] + [c]) &= \{a(b + c) \mid a \in [a], b \in [b], c \in [c]\}, \\ [a][b] + [a][c] &= \{a_1 b + a_2 c \mid a_1, a_2 \in [a], b \in [b], c \in [c]\}. \end{aligned}$$

■

Next we want to insert intervals into mappings.

**Definition 1.7 (mappings for intervals)** *Let  $G : \mathbb{R} \rightarrow \mathbb{R}$  be a real valued mapping and  $[a] \in \mathbb{I}$  an arbitrary interval. Then we define*

$$G([a]) := \{G(a) \mid a \in [a]\} \subset \mathbb{R}.$$

If  $G([a])$  is bounded we define

$$[G([a])] := [\min G([a]), \max G([a])] \subset \mathbb{R}.$$

The  $n$ -dimensional case is again defined componentwise. Let  $F = \begin{pmatrix} F_1 \\ \vdots \\ F_n \end{pmatrix} : \mathbb{R} \rightarrow \mathbb{R}^n$  be a  $n$ -dimensional function and  $[a] \in \mathbb{I}$  an interval. Then we define

$$F([a]) := \begin{pmatrix} F_1([a]) \\ \vdots \\ F_n([a]) \end{pmatrix} \subset \mathbb{R}^n.$$

If  $F([a])$  is bounded in all components, we define

$$[F([a])] := \begin{pmatrix} [F_1([a])] \\ \vdots \\ [F_n([a])] \end{pmatrix} \subset \mathbb{R}^n.$$

We can factor out a constant interval from a real integral.

**Lemma 1.8 (integration)** Let  $[a] \in \mathbb{I}$  be an interval and  $G : \mathbb{R} \rightarrow \mathbb{R}$  be a Lebesgue-integrable mapping. Then

$$\int [a]G(s) \, ds = [a] \int G(s) \, ds.$$

**Proof:** We calculate easily

$$\begin{aligned} \int [a]G(s) \, ds &= \left\{ \int a \, G(s) \, ds \mid a \in [a] \right\} \\ &= \left\{ a \int G(s) \, ds \mid a \in [a] \right\} \\ &= [a] \int G(s) \, ds. \end{aligned}$$

■

As a tool to increase the size of an interval by a relative amount we introduce the  $\varepsilon$ -inflation.

**Definition 1.9 ( $\varepsilon$ -inflation)** Let  $[a] \in \mathbb{I}^n$  be an interval. Then we define for  $\varepsilon > 0$

$$[a]_\varepsilon := (1 + \varepsilon)[a] - \varepsilon[a]$$

the  $\varepsilon$ -inflation of  $[a]$ .

**Lemma 1.10** Consider an interval  $[a] \in \mathbb{I}^n$ . Then the following inclusion holds true

$$[a] \subseteq [a]_\varepsilon.$$

**Proof:** For every component  $[a_i]$  of  $[a]$  we get with  $\underline{a} := \min[a_i]$  and  $\bar{a} := \max[a_i]$ , thus  $\underline{a} \leq \bar{a}$ :

$$\begin{aligned} \min[a_i]_\varepsilon &= (1 + \varepsilon)\underline{a} - \varepsilon\bar{a} = \underline{a} + \varepsilon(\underline{a} - \bar{a}) \leq \underline{a} \\ \max[a_i]_\varepsilon &= (1 + \varepsilon)\bar{a} - \varepsilon\underline{a} = \bar{a} + \varepsilon(\bar{a} - \underline{a}) \geq \bar{a} \\ \implies [a_i] &\subseteq [a_i]_\varepsilon, \text{ for all } i = 1, \dots, n \\ \implies [a] &\subseteq [a]_\varepsilon \end{aligned}$$

■

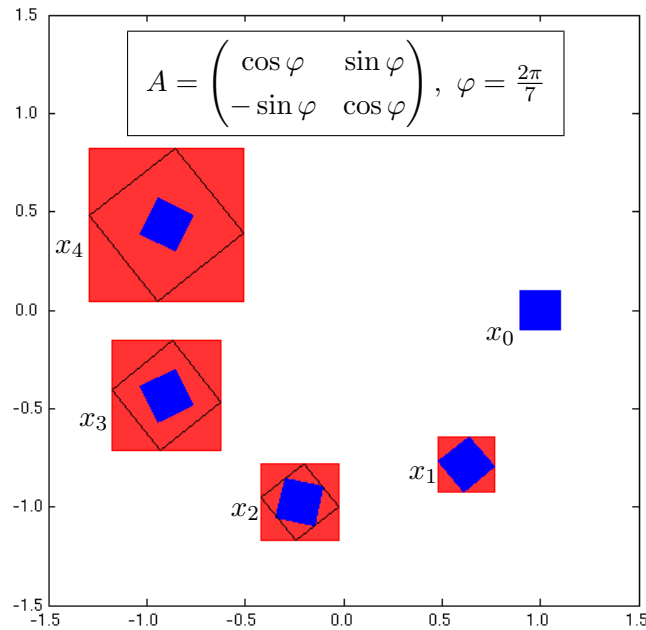
**Example 1.11**

$$[0.0, 1.0]_{0.1} = [0.0, 1.1] - [0.0, 0.1] = [-0.1, 1.1]$$

Now we will discuss a famous problem of interval arithmetic. It is the so called wrapping effect. To illustrate it we look at a two-dimensional example. Figure 1.1 shows a simple rotating map  $A \in \mathbb{R}^{2 \times 2}$ . We apply  $x_{i+1} = Ax_i$  recursively on an initial interval  $x_0$  and get a sequence  $x_0, x_1, x_2, \dots$  of sets. With interval arithmetics we compute in every iteration step an interval enclosure  $[x_{i+1}] = [A[x_i]]$  for the set  $A[x_i]$  (black box). So we overestimate the real result in every computation step. In the example the red interval  $[x_4]$  is much bigger than the exact result  $x_4 = A^4x_0$  (blue box).

One possibility to reduce the wrapping effect is the patching strategy. We subdivide the interval  $[x_0] = \bigcup_{k=1, \dots, n} [x_0]_k$  into a collection of intervals and execute the map  $[x_{i+1}]_k = [A[x_i]_k]$  for each of them separately. The result gets better, i.e.  $[x_i] \supset \bigcup_{k=1, \dots, n} [x_i]_k \supset x_i$ . For illustration compare figure 1.2. Of course, this strategy increases the numerical effort.



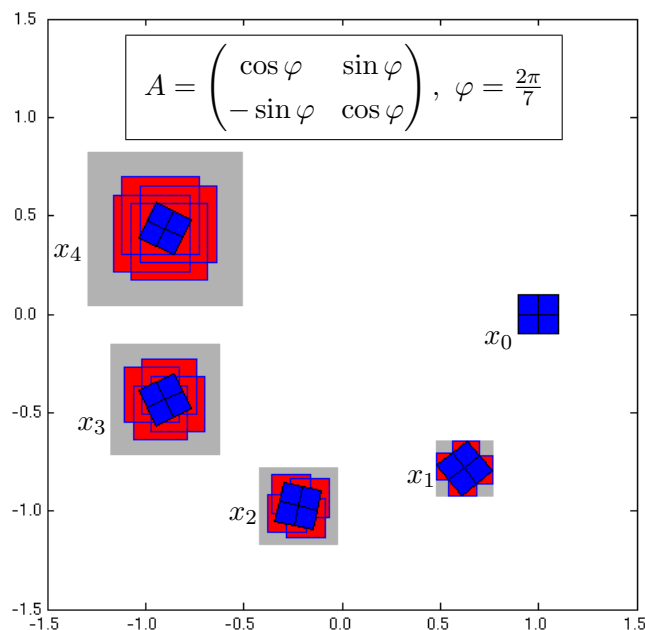


**Figure 1.1:** Illustration of the wrapping effect: The blue boxes  $x_0, \dots, x_4$  show the iteration  $x_{i+1} = Ax_i$ , a clockwise rotation, for the initial box  $x_0 = [-1.1, 0.9]$ . The red box  $[x_{i+1}]$  gives the smallest interval enclosure for  $A[x_i]$ , which is the black rotated rectangle.

Another idea to handle the wrapping effect is given by Lohner [26]. He recommended not to restrict the intervals to be parallel to the coordinate axes, but to represent a set with a rotated interval. In [27] Lohner gives an overview over recent strategies against the wrapping effect in various mathematical problem.

In computer sciences already a recent number of numerical tools and libraries for interval arithmetics exists. We refer at least to the tools used in this thesis. MATLAB users can easily get started with the toolbox INTLAB [31]. Hargreaves [13] gives a detailed introduction into interval arithmetic and provides a tutorial to learn INTLAB. He describes several basic applications like solving a linear equation and the interval Gaussian elimination. As a popular example of interval analysis an interval Newton algorithm for finding enclosures of all roots of a nonlinear system is discussed amongst others.

The C++ library C-XSC was developed and made available for academic use by Kulisch [22]. Now the group around Krämer and Hofschuster [15, 16]



**Figure 1.2:** Patching strategy: The initial interval  $x_0$  is subdivided in four intervals and the interval results of the iterative map  $[x_{i+1}]_k = [A[x_i]_k]$ ,  $k = 1, \dots, 4$  are given by the red boxes. The result is better (smaller) than the computation for big box (gray). The exact result is given by the blue boxes.

provides the C-XSC 2.0 class library. The C-XSC library is not only good for interval arithmetic. It is a huge library for scientific computing. It provides matrix and vector classes and represents real and complex numbers, of which the precision can be adjusted by the user. For these data types it provides a huge number of mathematical standard functions of high accuracy. Again as standard example the Interval Newton Method is introduced in the tutorial [16].

### 1.3 Multi-Index Theory

For the concatenation of indexed operators we define in this section multi-indices together with some useful manipulations. We will learn how a multi-index can be separated into two parts and how two multi-indices can be inserted into each other, while the internal sequence is preserved. We want to understand the shuffle product of two indices, which is defined recursively and can be expressed explicitly.

### 1.3.1 Definition

Reading from the right to the left one can see which operations of an indexed operator are executed consecutively. We define the multi-index set on the basis of the integer numbers and on the basis set  $\{0, \dots, m\}$ , where  $m$  will be the dimension of the control vector in system (1.1).

**Definition 1.12 (multi-indices)** *A multi-index is a row vector of nonnegative integer numbers with arbitrary dimension  $l \geq 0$ :*

$$b := (\beta_l, \dots, \beta_1) \in \mathbb{N}_0^l$$

*The empty index we denote by  $\ominus$ . We are interested in different sets of multi-indices:*

$$\mathcal{M}^l := \mathbb{N}_0^l = \{(\beta_l, \dots, \beta_1), \beta_i \in \mathbb{N}_0, 1 \leq i \leq l\},$$

$$\mathcal{M} := \bigcup_{l=0}^{\infty} \mathcal{M}^l = \{(\beta_l, \dots, \beta_1), l \geq 0, \beta_i \in \mathbb{N}_0, 1 \leq i \leq l\}.$$

*Now we restrict the index elements to the basis set  $\{0, \dots, m\}$ :*

$$\mathcal{M}_m^l := \{0, \dots, m\}^l = \{(\beta_l, \dots, \beta_1), \beta_i \in \{0, \dots, m\}, 1 \leq i \leq l\},$$

$$\mathcal{M}_m := \bigcup_{l=0}^{\infty} \mathcal{M}_m^l = \{(\beta_l, \dots, \beta_1), l \geq 0, \beta_i \in \{0, \dots, m\}, 1 \leq i \leq l\}.$$

One can easily assert the inclusions:

$$\begin{aligned} \mathcal{M}_m^l &\subset \mathcal{M}^l \subset \mathcal{M} \\ \mathcal{M}_m^l &\subset \mathcal{M}_m \subset \mathcal{M}. \end{aligned}$$

Next we define the length and the concatenation and separation operators for multi-indices:

**Definition 1.13** *Let  $b \in \mathcal{M}$  be an arbitrary multi-index  $b = (\beta_l, \dots, \beta_1)$ . Then we define the length as the number of indices of  $b$ :*

$$|b| := l. \tag{1.3}$$

With a second index  $a = (\alpha_{|a|}, \dots, \alpha_1) \in \mathcal{M}$  we define the concatenation

$$\begin{aligned} (\alpha, b) &:= (\alpha, \beta_l, \dots, \beta_1), \text{ for all } \alpha \in \mathbb{N}_0, \\ (a, b) &:= (\alpha_{|a|}, \dots, \alpha_1, \beta_{|b|}, \dots, \beta_1). \end{aligned}$$

The separation into right and left part, for  $k = 0, \dots, |b|$  is:

$$R_k(b) := (\beta_k, \dots, \beta_1), \quad (1.4)$$

$$L_k(b) := (\beta_{|b|}, \dots, \beta_{|b|-k+1}). \quad (1.5)$$

One can easily see, that the concatenation and the separation cancel out each other

$$\begin{aligned} b &= (L_{|b|-k}(b), R_k(b)), \text{ for all } k = 0, \dots, |b|, \\ a &= L_{|a|}((a, b)) \text{ and } b = R_{|b|}((a, b)). \end{aligned}$$

**Definition 1.14 (summation of multi-indices)**

For  $i \in \mathbb{N}$  and  $b_1, \dots, b_i, c \in \mathcal{M}$  we define the sum notation by

$$\left( \sum_{j=1}^i b_j, c \right) := \sum_{j=1}^i (b_j, c). \quad (1.6)$$

Its length is only well defined if the lengths of the indices on the right hand side coincide

$$|b_1| = \dots = |b_i| \iff \left| \sum_{j=1}^i b_j \right| := |b_1|.$$

**Definition 1.15 (hierarchical index-set and its remainder-set)** A finite index-set  $\mathcal{H} \subset \mathcal{M}_m$  is called hierarchical, if

$$\forall b \in \mathcal{H} \setminus \{\ominus\} \implies R_{|b|-1}(b) \in \mathcal{H}.$$

The remainder-set of a hierarchical index-set is defined by

$$\mathcal{R}(\mathcal{H}) := \{b \in \mathcal{M}_m \setminus \mathcal{H} \mid R_{|b|-1}(b) \in \mathcal{H}\}.$$

In particular, for every  $p \in \mathbb{N}_0$  the set  $\mathcal{G}_m^p := \bigcup_{i=0}^p \mathcal{M}_m^i$  is hierarchical. Its remainder set is  $\mathcal{R}(\mathcal{G}_m^p) = \mathcal{M}_m^{p+1}$ .

### 1.3.2 Shuffle-Product

Where the addition of multi-indices is defined in a canonical way, we will define in this section a concept for the multiplication. This notion was already given by Fliess [9] and is called “shuffle-product”. The term “shuffle” is taken from card-playing. There the dealer separates the pack of cards into two stacks and shuffles them into each other. The sequence of each stack is preserved. The sum over all possibilities to shuffle the cards in this way corresponds to the shuffle product on multi-indices. We define it recursively.

**Definition 1.16 (shuffle-product)** For  $b, c \in \mathcal{M}$  and  $\beta, \gamma \in \mathbb{N}_0$  we define:

$$\begin{aligned} b \sqcup \ominus &:= \ominus \sqcup b := b \\ (b, \beta) \sqcup (c, \gamma) &:= (b \sqcup (c, \gamma), \beta) + ((b, \beta) \sqcup c, \gamma) \end{aligned}$$

**Lemma 1.17 (symmetry of the shuffle-product)** Let  $b, c \in \mathcal{M}$  be arbitrary multi-indices. Then

$$b \sqcup c = c \sqcup b.$$

**Proof:** We prove by induction over the sum of the lengths  $L := |b| + |c|$ .

**Initial step**  $L = 0$ :

$$\ominus \sqcup \ominus = \ominus = \ominus \sqcup \ominus.$$

**Induction step**  $L - 1 \rightarrow L$ , for all  $L > 0$ : First we look at the trivial case, where  $b = \ominus$  and  $|c| = L$ :

$$\ominus \sqcup c = c = c \sqcup \ominus.$$

The same holds true for  $c = \ominus$  and  $|b| = L$ .

Otherwise there exist indices  $b', c' \in \mathcal{M}$  and  $\beta, \gamma \in \mathbb{N}_0$  with  $b = (b', \beta)$  and  $c = (c', \gamma)$ .

$$\begin{aligned} (b', \beta) \sqcup (c', \gamma) &= (b' \sqcup (c', \gamma), \beta) + ((b', \beta) \sqcup c', \gamma) \\ &= ((b', \beta) \sqcup c', \gamma) + (b' \sqcup (c', \gamma), \beta) \end{aligned}$$

With  $|(b', \beta)| + |c'| = |b'| + |(c', \gamma)| = L - 1$  we can apply the induction hypothesis

$$\begin{aligned} &= (c' \sqcup (b', \beta), \gamma) + ((c', \gamma) \sqcup b', \beta) \\ &= (c', \gamma) \sqcup (b', \beta). \end{aligned}$$

■

### 1.3.3 Combinatorial Selection

For developing an explicit version of the shuffle-product we now define an operator  $\mathcal{K}(b, k, i)$ . It selects  $k$  indices out of the multi-index  $b$ . Therefore we have  $\binom{|b|}{k}$  possibilities. The parameter  $i$  fixes one of the realizations.

**Definition 1.18 (combinatorial selection)** *Let  $b \in \mathcal{M}$  be an arbitrary multi-index and  $\beta \in \mathbb{N}_0$  a nonnegative integer number. Then we define for all integer numbers  $k, i$ , with  $0 \leq k \leq |b| + 1$  and  $1 \leq i \leq \binom{|b| + 1}{k}$  the selection operators recursively*

$$\mathcal{K}(\ominus, 0, 1) := \ominus, \quad \widehat{\mathcal{K}}(\ominus, 0, 1) := \ominus,$$

and with  $\sigma := \binom{|b|}{k-1}$ :

$$\mathcal{K}((b, \beta), k, i) := \begin{cases} (\mathcal{K}(b, k-1, i), \beta), & 1 \leq i \leq \sigma, \\ \mathcal{K}(b, k, i - \sigma), & \sigma < i \leq \binom{|b| + 1}{k}. \end{cases}$$

Analogously we define the complementary selection operator

$$\widehat{\mathcal{K}}((b, \beta), k, i) := \begin{cases} \widehat{\mathcal{K}}(b, k-1, i), & 1 \leq i \leq \sigma, \\ (\widehat{\mathcal{K}}(b, k, i - \sigma), \beta), & \sigma < i \leq \binom{|b| + 1}{k}. \end{cases}$$

Thereby for all  $j \in \mathbb{N}_0$  the degenerated binomial coefficients are  $\binom{j}{0} := 1$  and  $\binom{j}{-1} := 0$ .

**Remark 1.19** For arbitrary  $b \in \mathcal{M}$  the operators  $\mathcal{K}(b, k, i)$  and  $\widehat{\mathcal{K}}(b, k, i)$  are well defined with  $k$  and  $i$  in the ranges

$$0 \leq k \leq |b|, \quad 1 \leq i \leq \binom{|b|}{k}.$$

For the marginal value we get  $\sigma = \binom{|b| - 1}{k - 1}$ .

**Lemma 1.20 (simple properties of  $\mathcal{K}(\cdot, \cdot, \cdot)$  and  $\widehat{\mathcal{K}}(\cdot, \cdot, \cdot)$ )** For all  $b \in \mathcal{M}$ ,  $0 \leq k \leq |b|$  and  $1 \leq i \leq \binom{|b|}{k}$  the following assertions hold true:

$$\begin{aligned} \mathcal{K}(b, 0, 1) &= \ominus, & \widehat{\mathcal{K}}(b, 0, 1) &= b, \\ \mathcal{K}(b, |b|, 1) &= b, & \widehat{\mathcal{K}}(b, |b|, 1) &= \ominus, \\ |\mathcal{K}(b, k, i)| &= k, & |\widehat{\mathcal{K}}(b, k, i)| &= |b| - k. \end{aligned}$$

**Proof:** The assertions follow directly from definition 1.18. ■

For better understanding we look at a simple example:

**Example 1.21** Consider  $b = (\alpha, \beta, \gamma)$  as a multi-index of length 3.

$k = 0, i = 1$ :

$$\mathcal{K}(b, 0, 1) = \ominus$$

$k = 1, i = 1, \dots, \binom{3}{1}$ :

$$\begin{aligned} \mathcal{K}(b, 1, 1) &= (\mathcal{K}((\alpha, \beta), 0, 1), \gamma) = (\gamma) \\ \mathcal{K}(b, 1, 2) &= \mathcal{K}((\alpha, \beta), 1, 1) = (\mathcal{K}((\alpha), 0, 1), \beta) = (\beta) \\ \mathcal{K}(b, 1, 3) &= \mathcal{K}((\alpha, \beta), 1, 2) = \mathcal{K}((\alpha), 1, 1) = (\alpha) \end{aligned}$$

$k = 2, i = 1, \dots, \binom{3}{2}$ :

$$\begin{aligned} \mathcal{K}(b, 2, 1) &= (\mathcal{K}((\alpha, \beta), 1, 1), \gamma) = (\mathcal{K}((\alpha), 0, 1), \beta, \gamma) = (\beta, \gamma) \\ \mathcal{K}(b, 2, 2) &= (\mathcal{K}((\alpha, \beta), 1, 2), \gamma) = (\mathcal{K}((\alpha), 1, 1), \gamma) = (\alpha, \gamma) \\ \mathcal{K}(b, 2, 3) &= \mathcal{K}((\alpha, \beta), 2, 1) = (\alpha, \beta) \end{aligned}$$

$k = 3, i = 1$ :

$$\mathcal{K}(b, 3, 1) = b$$

The operator  $\widehat{\mathcal{K}}(\cdot, \cdot, \cdot)$  gives the corresponding complements.

$$\begin{array}{ll} \widehat{\mathcal{K}}(b, 0, 1) = b & \widehat{\mathcal{K}}(b, 2, 1) = (\alpha) \\ \widehat{\mathcal{K}}(b, 1, 1) = (\alpha, \beta) & \widehat{\mathcal{K}}(b, 2, 2) = (\beta) \\ \widehat{\mathcal{K}}(b, 1, 2) = (\alpha, \gamma) & \widehat{\mathcal{K}}(b, 2, 3) = (\gamma) \\ \widehat{\mathcal{K}}(b, 1, 3) = (\beta, \gamma) & \widehat{\mathcal{K}}(b, 3, 1) = \ominus \end{array}$$

By the following lemma we learn how to rearrange the set of multi-indices of a given length using the selection operators. We will use this later for changing the succession of summations.

**Lemma 1.22** Fix the length  $l \in \mathbb{N}_0$ . Then we get for all  $k = 0, \dots, l$  and  $i = 1, \dots, \binom{l}{k}$

$$\mathcal{M}_m^l = \left\{ \left( \mathcal{K}(b, k, i), \widehat{\mathcal{K}}(b, k, i) \right), b \in \mathcal{M}_m^l \right\}.$$

**Proof:** We prove the assertion by induction over the length  $l$ .

**Initial step  $l = 0$ :** The index-set  $\mathcal{M}_m^0$  includes only the empty index  $\ominus$ . So this case becomes trivial.

**Induction step  $l - 1 \rightarrow l$ :** We first look at the simple case  $k = 0$ : With  $\mathcal{K}(b, 0, 1) = \ominus$  and  $\widehat{\mathcal{K}}(b, 0, 1) = b$  (cp. lemma 1.20) it immediately follows

$$\left\{ \left( \mathcal{K}(b, 0, 1), \widehat{\mathcal{K}}(b, 0, 1) \right), b \in \mathcal{M}_m^l \right\} = \left\{ b, b \in \mathcal{M}_m^l \right\} = \mathcal{M}_m^l.$$

Next we analyse the case  $0 < k \leq l$  and  $1 \leq i \leq \binom{l-1}{k-1}$ :

$$\begin{aligned} \mathcal{M}_m^l &= \left\{ b, b \in \mathcal{M}_m^l \right\} \\ &= \left\{ (b', \beta), b' \in \mathcal{M}_m^{l-1}, \beta \in \{0, \dots, m\} \right\}. \end{aligned}$$

We move  $\beta$  to the  $k$ -th position in multi-index  $(b', \beta)$ :

$$= \left\{ (L_{k-1}(b'), \beta, R_{l-k}(b')), b' \in \mathcal{M}_m^{l-1}, \beta \in \{0, \dots, m\} \right\}.$$

We apply the induction hypothesis for  $\mathcal{M}_m^{l-1}$  and are now able to replace  $b'$  by  $(\mathcal{K}(b', k-1, i), \widehat{\mathcal{K}}(b', k-1, i))$ , where  $|\mathcal{K}(b', k-1, i)| = k-1$  and



$\left| \widehat{\mathcal{K}}(b', k-1, i) \right| = l - k$  (cp. lemma 1.20). For the left and right parts we get then:

$$\begin{aligned} L_{k-1} \left( \left( \mathcal{K}(b', k-1, i), \widehat{\mathcal{K}}(b', k-1, i) \right) \right) &= \mathcal{K}(b', k-1, i), \\ R_{l-k} \left( \left( \mathcal{K}(b', k-1, i), \widehat{\mathcal{K}}(b', k-1, i) \right) \right) &= \widehat{\mathcal{K}}(b', k-1, i). \end{aligned}$$

Then we get

$$\begin{aligned} \mathcal{M}_m^l &= \\ &= \left\{ \left( \mathcal{K}(b', k-1, i), \beta, \widehat{\mathcal{K}}(b', k-1, i) \right), b' \in \mathcal{M}_m^{l-1}, \beta \in \{0, \dots, m\} \right\} \end{aligned}$$

and can apply definition 1.18 of  $\mathcal{K}(\cdot, \cdot, \cdot)$  and  $\widehat{\mathcal{K}}(\cdot, \cdot, \cdot)$

$$\begin{aligned} &= \left\{ \left( \mathcal{K}((b', \beta), k, i), \widehat{\mathcal{K}}((b', \beta), k, i) \right), b' \in \mathcal{M}_m^{l-1}, \beta \in \{0, \dots, m\} \right\} \\ &= \left\{ \left( \mathcal{K}(b, k, i), \widehat{\mathcal{K}}(b, k, i) \right), b \in \mathcal{M}_m^l \right\}. \end{aligned}$$

The remaining case  $0 < k \leq l$  and  $\binom{l-1}{k-1} < i \leq \binom{l}{k}$  can be proved analogously, but there the index  $\beta$  has not to be moved within the multi-index. ■

### 1.3.4 Insertion Operator

Now we define the insertion operator. It is the counterpart to the [selection operator](#), which selects a partial index out of a given multi-index. The insertion operator inserts two multi-indices into each other, where the internal order of each index is conserved.

**Definition 1.23 (combinatorial insertion operator)** *We consider two multi-indices  $c, d \in \mathcal{M}$ , two integer numbers  $\gamma, \delta \in \mathbb{N}_0$  and a nonzero integer number  $i \in \mathbb{N}$ , with  $1 \leq i \leq \Sigma := \binom{|c| + |d| + 2}{|c| + 1}$ . Then we define recursively*

$$\mathcal{A}(c, \ominus, 1) := c, \quad \mathcal{A}(\ominus, d, 1) := d$$

$$\text{and with } \sigma := \binom{|c| + |d| + 1}{|c|}$$

$$\mathcal{A}((c, \gamma), (d, \delta), i) := \begin{cases} (\mathcal{A}(c, (d, \delta), i), \gamma), & 1 \leq i \leq \sigma, \\ (\mathcal{A}((c, \gamma), d, i - \sigma), \delta), & \sigma < i \leq \Sigma. \end{cases}$$

**Remark 1.24** For arbitrary multi-indices  $c, d \in \mathcal{M}$  the operator  $\mathcal{A}(c, d, i)$  is well defined with

$$1 \leq i \leq \Sigma := \binom{|c| + |d|}{|c|}$$

and the marginal value

$$\sigma := \binom{|c| + |d| - 1}{|c| - 1}.$$

The following theorem gives us an explicit representation for the recursively defined [shuffle product](#). With  $C(l, k)$  we denote the binomial coefficient  $\binom{l}{k}$ .

**Theorem 1.25 (explicit shuffle product)** Let  $c, d \in \mathcal{M}$  be two arbitrary multi-indices. Then their shuffle product can be explicitly represented by

$$c \sqcup d = \sum_{i=1}^{C(|c|+|d|, |c|)} \mathcal{A}(c, d, i).$$

**Proof:** We prove by induction over the common length  $L := |c| + |d|$ .

**Initial step**  $L = 0$ :

$$\ominus \sqcup \ominus = \ominus = \mathcal{A}(\ominus, \ominus, 1).$$

**Induction step**  $L - 1 \rightarrow L$ : First we look at the two trivial cases  $L = |c|$  and  $L = |d|$

$$c \sqcup \ominus = c = \mathcal{A}(c, \ominus, 1),$$

$$\ominus \sqcup d = d = \mathcal{A}(\ominus, d, 1),$$

and second we analyse the case  $|c| > 0$  and  $|d| > 0$ . Here exist indices  $c', d' \in \mathcal{M}$  and integers  $\gamma, \delta \in \mathbb{N}_0$  with  $c = (c', \gamma)$  and  $d = (d', \delta)$ . They follow the recursive definition of the shuffle product [1.16](#)

$$\begin{aligned} c \sqcup d &= (c', \gamma) \sqcup (d', \delta) \\ &= ((c', \gamma) \sqcup d', \delta) + (c' \sqcup (d', \delta), \gamma) \\ &= (c \sqcup d', \delta) + (c' \sqcup d, \gamma). \end{aligned}$$

We apply the induction hypothesis

$$= \sum_{i=1}^{C(|c|+|d|-1,|c|)} (\mathcal{A}(c, d', i), \delta) + \sum_{i=1}^{C(|c|+|d|-1,|c|-1)} (\mathcal{A}(c', d, i), \gamma).$$

With the combinatorial formula  $\binom{l}{k} = \binom{l-1}{k} + \binom{l-1}{k-1}$ , for all  $l, k \in \mathbb{N}$ , we get

$$\begin{aligned} &= \sum_{i=1+C(|c|+|d|-1,|c|-1)}^{C(|c|+|d|,|c|)} (\mathcal{A}(c, d', i - C(|c|+|d|-1,|c|-1)), \delta) \\ &\quad + \sum_{i=1}^{C(|c|+|d|-1,|c|-1)} (\mathcal{A}(c', d, i), \gamma). \end{aligned}$$

From definition 1.23 follows the assertion

$$= \sum_{i=1}^{C(|c|+|d|,|c|)} \mathcal{A}(c, d, i).$$

■

Now we analyse the relation between the selection and insertion operators.

**Theorem 1.26 (duality of selection and insertion operator)**

For all  $b \in \mathcal{M}$ ,  $k = 0, \dots, |b|$  and  $i = 1, \dots, \binom{|b|}{k}$  it holds true:

$$\mathcal{A}(\mathcal{K}(b, k, i), \widehat{\mathcal{K}}(b, k, i), i) = b$$

**Proof:** We prove by induction over the length  $L := |b|$ :

**Initial step**  $L = 0$ :

$$\mathcal{A}(\mathcal{K}(\ominus, 0, 1), \widehat{\mathcal{K}}(\ominus, 0, 1), 1) = \mathcal{A}(\ominus, \ominus, 1) = \ominus.$$

**Induction step**  $L - 1 \rightarrow L$ :

First we look at the two trivial cases  $k = 0$  and  $k = L$  (cp. lemma 1.20).

$$\begin{aligned}\mathcal{A}\left(\mathcal{K}(b, 0, 1), \widehat{\mathcal{K}}(b, 0, 1), 1\right) &= \mathcal{A}(\ominus, b, 1) = b \\ \mathcal{A}\left(\mathcal{K}(b, L, 1), \widehat{\mathcal{K}}(b, L, 1), 1\right) &= \mathcal{A}(b, \ominus, 1) = b.\end{aligned}$$

Second we analyse the case  $0 < k < L$  and  $1 \leq i \leq \binom{L-1}{k-1}$ . There exist  $b' \in \mathcal{M}$  and  $\beta \in \mathbb{N}_0$  with  $b = (b', \beta)$ .

$$\begin{aligned}\mathcal{A}\left(\mathcal{K}(b, k, i), \widehat{\mathcal{K}}(b, k, i), i\right) &= \\ &= \mathcal{A}\left(\mathcal{K}((b', \beta), k, i), \widehat{\mathcal{K}}((b', \beta), k, i), i\right) \\ &= \mathcal{A}\left(\mathcal{K}(b', k-1, i), \widehat{\mathcal{K}}(b', k-1, i), i\right) \\ &= \left(\mathcal{A}\left(\mathcal{K}(b', k-1, i), \widehat{\mathcal{K}}(b', k-1, i), i\right), \beta\right).\end{aligned}$$

We apply the induction hypothesis.

$$= (b', \beta) = b$$

The last case,  $0 < k < L$  and  $\binom{L-1}{k-1} < i \leq \binom{L}{k}$ , we treat analogously, where  $\sigma := \binom{L-1}{k-1}$ .

$$\begin{aligned}\mathcal{A}\left(\mathcal{K}(b, k, i), \widehat{\mathcal{K}}(b, k, i), i\right) &= \\ &= \mathcal{A}\left(\mathcal{K}((b', \beta), k, i), \widehat{\mathcal{K}}((b', \beta), k, i), i\right) \\ &= \mathcal{A}\left(\mathcal{K}(b', k, i-\sigma), \widehat{\mathcal{K}}(b', k, i-\sigma), i\right) \\ &= \left(\mathcal{A}\left(\mathcal{K}(b', k, i-\sigma), \widehat{\mathcal{K}}(b', k, i-\sigma), i-\sigma\right), \beta\right) \\ &= (b', \beta) = b.\end{aligned}$$

■

### 1.3.5 Serial Number Representation

In applications the multi-index set  $\mathcal{M}_m$  is used to name coefficients. It is a countable set so it can be identified with serial numbers, which is the set of nonnegative integer numbers  $\mathbb{N}_0$ . In implementations it is easier to handle serial numbers than indices. Therefore we define in this section operators to

map from  $\mathcal{M}_m$  to  $\mathbb{N}_0$  and vice versa and we define the counterpart for the selection operator acting on serial numbers. We begin with the definition of the operator  $L$  to compute serial numbers out of multi-indices.

**Definition 1.27** For  $b \in \mathcal{M}_m$  and  $\beta \in \{0, \dots, m\}$  we define the operator  $L : \mathcal{M}_m \rightarrow \mathbb{N}_0$  recursively by

$$\begin{aligned} L[\ominus] &:= 0, \\ L[(b, \beta)] &:= L[b](m+1) + \beta + 1. \end{aligned}$$

And we define the operator  $M$  to get the multi-index out of a serial number analogously.

**Definition 1.28** For  $l \in \mathbb{N}_0$  we define the operator  $M : \mathbb{N}_0 \rightarrow \mathcal{M}_m$  recursively by

$$M[l] := \begin{cases} \ominus, & \text{for } l = 0, \\ (M[(l-1) \setminus (m+1)], (l-1) \bmod (m+1)), & \text{otherwise.} \end{cases}$$

Thereby the operator “ $\setminus$ ” denotes the integer division and “ $\bmod$ ” denotes its remainder.

**Definition 1.29** We define the length of an integer index  $l \in \mathbb{N}_0$  by the length of its corresponding multi-index:

$$|l| := |M[l]|.$$

The length  $|l|$  can be easily computed without explicitly computing  $M[l]$ :

$$|l| = \begin{cases} 0, & \text{for } l = 0, \\ |(l-1) \setminus (m+1)| + 1, & \text{otherwise.} \end{cases} \quad (1.7)$$

**Lemma 1.30** The operators  $L$  and  $M$  are inverse, that is for all  $b \in \mathcal{M}_m$  and all  $l \in \mathbb{N}_0$

$$L[M[l]] = l, \quad M[L[b]] = b.$$

**Proof:** Obviously the integer division and its remainder satisfy the equation

$$l - 1 = [(l-1) \setminus (m+1)](m+1) + (l-1) \bmod (m+1).$$

And we use the recursive formula for the length of  $l > 0$  (cp. equation 1.7):

$$|l| - 1 = |(l-1) \setminus (m+1)|.$$

We prove the first assertion of lemma 1.30 by induction over the length  $|l|$ .

**Initial step**  $|l| = 0$ :

$$L[M[0]] = L[\ominus] = 0.$$

**Induction step**  $|l| - 1 \rightarrow |l|$ : We apply the definitions 1.28 and 1.27

$$\begin{aligned} L[M[l]] &= L[(M[(l-1) \setminus (m+1)], (l-1) \bmod (m+1))] \\ &= L[M[(l-1) \setminus (m+1)] \cdot (m+1) + (l-1) \bmod (m+1) + 1, \end{aligned}$$

by using the induction hypothesis we get

$$\begin{aligned} &= ((l-1) \setminus (m+1)) \cdot (m+1) + (l-1) \bmod (m+1) + 1 \\ &= l - 1 + 1 = l. \end{aligned}$$

The second assertion we prove by induction, too. Therefore we recall that for every  $\beta \in \{0, \dots, m\}$  the integer division gives  $\beta \setminus (m+1) = 0$  with the remainder  $\beta \bmod (m+1) = \beta$ . Let be  $b \in \mathcal{M}_m$  a multi-index with length  $|b|$ .

**Initial step**  $|b| = 0$ :

$$M[L[\ominus]] = L[0] = \ominus.$$

**Induction step**  $|b| \rightarrow |b| + 1$ : We apply the definitions 1.27 and 1.28 and then the induction hypothesis.

$$\begin{aligned} M[L[(b, \beta)]] &= M[L[b] \cdot (m+1) + \beta + 1] \\ &= (M[(L[b] \cdot (m+1) + \beta) \setminus (m+1)], (L[b] \cdot (m+1) + \beta) \bmod (m+1)) \\ &= (M[L[b]], \beta) \\ &= (b, \beta). \end{aligned}$$

■

From our previously defined operators on multi-indices the selection operator is the only one we use for the implementation, namely for the product formula of Fliess-expansions (cp. theorem 2.33), which we will develop in chapter 2. The shuffle-product and the selection operator are matters of theoretical interest.

We define the selection operator of a serial number by the selection operator of its corresponding multi-index.

**Definition 1.31** For all  $l \in \mathbb{N}_0$ ,  $k = 0, \dots, |l|$  and  $i = 1, \dots, \binom{|l|}{k}$  we define the selection operators on serial numbers by:

$$\begin{aligned} K(l, k, i) &:= L[\mathcal{K}(M[l], k, i)], \\ \widehat{K}(l, k, i) &:= L[\widehat{\mathcal{K}}(M[l], k, i)]. \end{aligned}$$

From the definitions 1.18 and 1.31 we immediately get the trivial case  $K(0, 0, 1) = 0$  and  $\widehat{K}(0, 0, 1) = 0$ . For  $l > 0$  we formulate the following lemma. In the following lemma we give recursive formulas for the selection operator on serial numbers without the detour on multi-indices.

**Lemma 1.32** For all  $l \in \mathbb{N}$  we get with  $k = 0, \dots, |l|$ ,  $i = 1, \dots, \binom{|l|}{k}$  and the marginal value  $\sigma := \binom{|l| - 1}{k - 1}$ :

$$\begin{aligned} K(l, k, i) &= \begin{cases} K(D, k - 1, i) \cdot (m + 1) + R + 1, & 1 \leq i \leq \sigma, \\ K(D, k, i - \sigma), & \sigma < i \leq \binom{|l|}{k}, \end{cases} \\ \widehat{K}(l, k, i) &= \begin{cases} \widehat{K}(D, k - 1, i), & 1 \leq i \leq \sigma, \\ \widehat{K}(D, k, i - \sigma) \cdot (m + 1) + R + 1, & \sigma < i \leq \binom{|l|}{k}, \end{cases} \end{aligned}$$

where  $D := (l - 1) \setminus (m + 1)$  and  $R := (l - 1) \bmod (m + 1)$ .

**Proof:** We will only prove the first assertion of lemma 1.32. The assertion for the complementary selection operator can be proved analogously. We start with definition 1.31 and apply definition 1.28

$$K(l, k, i) = L[\mathcal{K}(M[l], k, i)] = L[\mathcal{K}((M[D], R), k, i)].$$

Now we use the definition 1.18 of the selection operator

$$K(l, k, i) = \begin{cases} L[\mathcal{K}(M[D], k - 1, i), R], & 1 \leq i \leq \sigma, \\ L[\mathcal{K}(M[D], k, i - \sigma)], & \sigma < i \leq \binom{|l|}{k}. \end{cases} \quad (1.8)$$

The first case we simplify with definition 1.27

$$= \begin{cases} L[\mathcal{K}(M[D], k - 1, i) \cdot (m + 1) + R + 1, & 1 \leq i \leq \sigma, \\ L[\mathcal{K}(M[D], k, i - \sigma)], & \sigma < i \leq \binom{|l|}{k}, \end{cases}$$

what becomes with definition [1.31](#)

$$= \begin{cases} K(D, k-1, i) \cdot (m+1) + R+1, & 1 \leq i \leq \sigma, \\ K(D, k, i-\sigma), & \sigma < i \leq \binom{|l|}{k}. \end{cases}$$

■



## Chapter 2

# Fliess-Expansions

In this chapter we introduce Fliess-expansions. They are a generalisation of Taylor-expansions for solutions of ordinary differential equations on control affine systems. They depend on the time variable and the vector of control functions.

First we will define the Lie-derivatives which gives the derivatives of the solution in direction of the right hand side vector fields. The Lie-derivatives of the solution at the initial conditions basically give the coefficients of its Fliess-expansion.

The Fliess-expansion is a higher order integral representation of the initial value problem for system (1.1). For a readable notation we define the iterated integral operator. It integrates a given kernel multiplied with control functions determined by a given multi-index on an high-dimensional time triangle. We will discover many properties of iterated integrals in order to perform the basic arithmetic operations on Fliess-expansions. Especially we develop multiplication formulas in quantitative and qualitative versions for iterated integrals.

We will give a new proof for the uniqueness of Fliess-expansions. It was already proved by Fliess [9]. He used an algebraic approach, which is hard to understand in our context and notation. So we use his tools, i.e. the Fliess-derivative, but in terms of analysis. The Fliess-derivative allows us to isolate any coefficient of the Fliess-expansion. Then we will be able to compare Fliess-expansions coefficientwise.

We will learn to represent other functions than a solution as Fliess-expansion, i.e. the right hand side vector fields or control independent functions, which are smooth enough. We develop the basic arithmetic operations for Fliess-expansions. That allows us to compute new expansions out of the solution in order to get a Fliess representation of the right hand side vector

fields.

## 2.1 Lie derivatives

The following definition for Lie derivatives is taken from Grüne, Kloeden [12]. First they are defined for variables in time and state space. Later we will insert an arbitrary time dependent solution.

**Definition 2.1 (Lie derivatives)** *Let  $F : [\tau, T] \times \mathbb{R}^n \longrightarrow \mathbb{R}^n$  be continuously differentiable. Then we define the linear operator  $\mathcal{L}^0 F(t, x) := \frac{dF}{dt}(t, x)$  as the derivative of  $F$  in the first variable. For  $\alpha \in \{1, \dots, m\}$  the linear operator  $\mathcal{L}^\alpha F(t, x) := \frac{dF}{dx}(t, x) f^\alpha(t, x)$  is the derivative of  $F$  in direction of the vector field  $f^\alpha$ .*

We insert the independent time variable  $t$  and the function  $x : [\tau, T] \rightarrow \mathbb{R}^n$ , which now denotes an arbitrary solution of the control system 1.1 for a fixed vector of control functions  $u \in \mathcal{U}^m$ , and get a function  $F(\cdot, x(\cdot)) : [\tau, T] \rightarrow \mathbb{R}^n$ . The next lemma extends the First Fundamental Theorem of Calculus for control affine systems.

**Lemma 2.2** *For all  $t \in [\tau, T]$  and any solution  $x$  we have*

$$F(t, x(t)) = F(\tau, x(\tau)) + \int_{\tau}^t \sum_{\alpha=0}^m \mathcal{L}^\alpha F(s, x(s)) u^\alpha(s) ds. \quad (2.1)$$

**Proof:** From the chain rule and the [solution identity](#) follows:

$$\begin{aligned} \frac{d}{ds} F(s, x(s)) &= \frac{dF}{dt}(s, x(s)) + \frac{dF}{dx}(s, x(s)) \dot{x}(s) \\ &= \frac{dF}{dt}(s, x(s)) + \sum_{\alpha=1}^m \frac{dF}{dx}(s, x(s)) f^\alpha(s, x(s)) u^\alpha(s) \\ &= \mathcal{L}^0 F(s, x(s)) + \sum_{\alpha=1}^m \mathcal{L}^\alpha F(s, x(s)) u^\alpha(s) \\ &= \sum_{\alpha=0}^m \mathcal{L}^\alpha F(s, x(s)) u^\alpha(s), \quad \text{where } u^0 \equiv 1 \end{aligned}$$

Then the proof is given by the First Fundamental Theorem of Calculus. ■

For  $F(t, x) = x$  we get the integral equation corresponding to system 1.1:

$$x(t) = x(\tau) + \int_{\tau}^t \sum_{\alpha=0}^m \mathcal{L}^{\alpha} x(s) u^{\alpha}(s) \, ds. \quad (2.2)$$

By  $\mathcal{L}^{\alpha} x$  we denote the  $\mathcal{L}^{\alpha}$ -operator applied to the identity  $F(t, x) = x$ . By  $\mathcal{L}^{\alpha} x(s)$  we denote the evaluation at  $(s, x(s))$ . The empty index defines the identity  $\mathcal{L}^{\emptyset} x := x$ .

We abbreviate the concatenation of Lie derivatives with the multi-index notation. For any  $b = (\beta_{|b|}, \dots, \beta_1) \in \mathcal{M}_m$  we write:

$$\mathcal{L}^b = \mathcal{L}^{\beta_{|b|}} \circ \dots \circ \mathcal{L}^{\beta_1}.$$

For  $F(\cdot, x) := \mathcal{L}^b x$  equation (2.1) leads to

$$\mathcal{L}^b x(t) = \mathcal{L}^b x(\tau) + \int_{\tau}^t \sum_{\alpha=0}^m \mathcal{L}^{(\alpha, b)} x(s) u^{\alpha}(s) \, ds. \quad (2.3)$$

**Lemma 2.3** *For every  $b \in \mathcal{M}_m$  and  $\alpha \in \{0, \dots, m\}$  we get as the Lie-derivatives of the solution:*

$$\mathcal{L}^{(b, \alpha)} x = \begin{cases} 0, & \text{for } \alpha = 0, \\ \mathcal{L}^b f^{\alpha}, & \text{for } \alpha \in \{1, \dots, m\}. \end{cases}$$

**Proof:** For the identity  $F(t, x) = x$  the  $\mathcal{L}^0$ -operator gives  $\mathcal{L}^0 x = 0$ , because  $F$  is independent of the first variable. This propagates to the concatenation of Lie-derivatives. For  $\alpha \in \{1, \dots, m\}$  we get  $\mathcal{L}^{\alpha} x = f^{\alpha}$  immediately. ■

Note that from the point of view of lemma 2.3 the Lie-derivatives of the solution depend on the solution itself only indirectly through the vector fields  $f^{\alpha}$ ,  $\alpha \in \{1, \dots, m\}$ .

To prevent mistakes we recall the meaning of  $\mathcal{L}^b x : [\tau, T] \times \mathbb{R}^n \longrightarrow \mathbb{R}^n$ . It depends only on the right hand side vector fields, but is independent from the solution  $x$ . Here  $x$  is used as the state space variable. On the other hand we denote with  $\mathcal{L}^b x(t)$  the evaluation of  $\mathcal{L}^b x$  at  $(t, x(t))$ . In this case  $x$  is used as the solution.

## 2.2 Iterated Integrals

In this section we introduce another operator, which uses our concept of multi-indices for concatenation. The iterated integral is a multi-integral, that includes a scalar control function determined by the multi-index in every iteration. Again the definition is taken from Grüne, Kloeden [12].

**Definition 2.4** Denote with  $G : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  a Lebesgue-integrable function which depends on the vector of control functions. For all multi-indices  $b \in \mathcal{M}_m$  and  $\alpha \in \{0, \dots, m\}$  we define the iterated integral recursively by

$$\begin{aligned} \mathcal{I}_\ominus [G] (t, u) &:= G(t, u) \\ \mathcal{I}_{(\alpha, b)} [G] (t, u) &:= \mathcal{I}_b \left[ \int_\tau^\cdot G(s, u) u^\alpha(s) \, ds \right] (t, u), \\ &\text{where } u^0 \equiv 1 \end{aligned} \quad (2.4)$$

For the identity integral kernel we abbreviate

$$\mathcal{I}_b(t, u) := \mathcal{I}_b[1](t, u).$$

**Remark 2.5** In fact the vector of control functions  $u = (u^1, \dots, u^m) \in \mathcal{U}^m$  should include the 0-component. But by convention we need not extend  $u$  for the constant component  $u^0 \equiv 1$ .

The representation for indices  $b$  of length  $|b| = 1, 2, 3$  can be computed by recursive insertion. For  $\alpha, \beta, \gamma \in \{0, \dots, m\}$  we get

$$\begin{aligned} \mathcal{I}_{(\gamma)} [G] (t, u) &= \int_\tau^t G(s_1, u) u^\gamma(s_1) \, ds_1 \\ \mathcal{I}_{(\beta, \gamma)} [G] (t, u) &= \int_\tau^t \int_\tau^{s_1} G(s_2, u) u^\beta(s_2) \, ds_2 \, u^\gamma(s_1) \, ds_1 \\ \mathcal{I}_{(\alpha, \beta, \gamma)} [G] (t, u) &= \int_\tau^t \int_\tau^{s_1} \int_\tau^{s_2} G(s_3, u) u^\alpha(s_3) \, ds_3 \, u^\beta(s_2) \, ds_2 \, u^\gamma(s_1) \, ds_1. \end{aligned}$$

Continuing this procedure for multi-indices  $b = (\beta_{|b|}, \dots, \beta_1) \in \mathcal{M}_m$  of arbitrary length leads to

$$\begin{aligned} \mathcal{I}_{(\beta_{|b|}, \dots, \beta_1)} [G] (t, u) &= \\ &= \int_\tau^t \int_\tau^{s_1} \dots \int_\tau^{s_{|b|-1}} G(s_{|b|}, u) u^{\beta_{|b|}}(s_{|b|}) \, ds_{|b|} \dots u^{\beta_2}(s_2) \, ds_2 \, u^{\beta_1}(s_1) \, ds_1. \end{aligned} \quad (2.5)$$

Next we formulate several lemmas which help us to deal with iterated integrals. Because control functions are not continuous in general, the iterated integrals are differentiable only once and the derivative is not continuous.

**Lemma 2.6 (derivative of iterated integrals)** *Consider a multi-index  $b \in \mathcal{M}_m$  and an index  $\alpha \in \{0, \dots, m\}$  and let  $G : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  be a integrable function. Then the time derivatives of the iterated integrals are given by*

$$\frac{d}{dt} \mathcal{I}_{(b, \alpha)} [G] (t, u) = \mathcal{I}_b [G] (t, u) u^\alpha(t).$$

**Proof:** We prove by induction on the length  $L := |b|$ .

**Initial step:**  $L = 0$ , thus  $b = \ominus$ :

$$\frac{d}{dt} \mathcal{I}_{(\alpha)} [G] (t, u) = \frac{d}{dt} \mathcal{I}_{\ominus} \left[ \int_{\tau}^{\cdot} G(s, u) u^\alpha(s) ds \right] (t, u) = G(t, u) u^\alpha(t)$$

**Induction step:**  $L - 1 \rightarrow L, L \geq 1$ : There exists a multi index  $b' \in \mathcal{M}_m$  and an integer number  $0 \leq \beta \leq m$  with  $b = (\beta, b')$ .

$$\begin{aligned} \frac{d}{dt} \mathcal{I}_{(b, \alpha)} [G] (t, u) &= \frac{d}{dt} \mathcal{I}_{(\beta, b', \alpha)} [G] (t, u) \\ &= \frac{d}{dt} \mathcal{I}_{(b', \alpha)} \left[ \int_{\tau}^{\cdot} G(s, u) u^\beta(s) ds \right] (t, u) \end{aligned}$$

With  $|b'| = L - 1$  we can apply the induction hypothesis for  $\mathcal{I}_{(b', \alpha)}$

$$\begin{aligned} &= \mathcal{I}_{b'} \left[ \int_{\tau}^{\cdot} G(s, u) u^\beta(s) ds \right] (t, u) u^\alpha(t) \\ &= \mathcal{I}_{(\beta, b')} [G] (t, u) u^\alpha(t) = \mathcal{I}_b [G] (t, u) u^\alpha(t) \end{aligned}$$

■

The derivative for iterated integrals gives us the possibility to extend the multi-index from the right hand side. We get a more convenient iteration than the definition of the iterated integrals 2.4.

**Lemma 2.7** *For the extension from the right of the multi-index  $b \in \mathcal{M}_m$  of an iterated integral by  $\alpha \in \{0, \dots, m\}$  we get*

$$\mathcal{I}_{(b, \alpha)} [G] (t, u) = \int_{\tau}^t \mathcal{I}_b [G] (s, u) u^\alpha(s) ds.$$

**Proof:** The assertion follows directly from the First Fundamental Theorem of Calculus by using lemma 2.6

$$\begin{aligned} \mathcal{I}_{(b, \alpha)} [G] (t, u) &= \mathcal{I}_{(b, \alpha)} [G] (\tau, u) + \int_{\tau}^t \frac{d}{ds} \mathcal{I}_{(b, \alpha)} [G] (s, u) ds \\ &= \int_{\tau}^t \mathcal{I}_b [G] (s, u) u^\alpha(s) ds. \end{aligned}$$

■

For an easier denotation of the sum of iterated integrals we move the sum into the index if the integral kernels coincide.

**Definition 2.8 (sum of iterated integrals)** *Consider multi-indices  $b, c \in \mathcal{M}_m$  and a integrable function  $G : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$ . Then we define the iterated integral for the sum of multi-indices as the sum of the iterated integrals for each of the multi-indices*

$$\mathcal{I}_{b+c}[G](t, u) := \mathcal{I}_b[G](t, u) + \mathcal{I}_c[G](t, u).$$

**Lemma 2.9** *For all  $b, c \in \mathcal{M}_m$ ,  $\alpha \in \{0, \dots, m\}$  and  $G : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  we get the following summation formula for iterated integrals*

$$\mathcal{I}_{(b+c, \alpha)}[G](t, u) = \mathcal{I}_{(b, \alpha)}[G](t, u) + \mathcal{I}_{(c, \alpha)}[G](t, u).$$

**Proof:** We apply lemma 2.7 in both directions.

$$\begin{aligned} \mathcal{I}_{(b+c, \alpha)}[G](t, u) &= \int_{\tau}^t \mathcal{I}_{b+c}[G](s, u) u^{\alpha}(s) ds \\ &= \int_{\tau}^t \mathcal{I}_b[G](s, u) u^{\alpha}(s) ds + \int_{\tau}^t \mathcal{I}_c[G](s, u) u^{\alpha}(s) ds \\ &= \mathcal{I}_{(b, \alpha)}[G](t, u) + \mathcal{I}_{(c, \alpha)}[G](t, u). \end{aligned}$$

■

**Lemma 2.10 (linearity of iterated integrals)** *The iterated integral operator is linear in the integral kernel.*

**Proof:** To show linearity we verify for all multi-indices  $b \in \mathcal{M}_m$ , all integrable functions  $G, H : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  and all numbers  $n_1, n_2 \in \mathbb{R}$ :

$$n_1 \mathcal{I}_b[G](t, u) + n_2 \mathcal{I}_b[H](t, u) = \mathcal{I}_b[n_1 G + n_2 H](t, u).$$

We prove by induction over the length  $L := |b|$ .

**Initial step:**  $L = 0$ , thus  $b = \ominus$ : This case is trivial.

**Induction step:**  $L - 1 \rightarrow L$ : There exist  $b' \in \mathcal{M}_m$  and  $\beta \in \{0, \dots, m\}$  with  $b = (b', \beta)$ . We apply lemma 2.7 twice and add up the iterated integrals using the induction hypothesis

$$\begin{aligned} n_1 \mathcal{I}_{(b', \beta)}[G](t, u) + n_2 \mathcal{I}_{(b', \beta)}[H](t, u) &= \\ &= \int_{\tau}^t n_1 \mathcal{I}_{b'}[G](s, u) u^{\beta}(s) ds + \int_{\tau}^t n_2 \mathcal{I}_{b'}[H](s, u) u^{\beta}(s) ds \\ &= \int_{\tau}^t \mathcal{I}_{b'}[n_1 G + n_2 H](s, u) u^{\beta}(s) ds \\ &= \mathcal{I}_{(b', \beta)}[n_1 G + n_2 H](t, u). \end{aligned}$$

■

The next lemma will analyse the case, where the integral kernel of an iterated integral is again an iterated integral.

**Lemma 2.11** *Let  $b, c \in \mathcal{M}_m$  be multi-indices and  $G : [\tau, T] \times \mathcal{U}^m \rightarrow \mathbb{R}^n$  be a function depending on the control functions. Then inserting one iterated integral into another is the same as concatenating their multi-indices.*

$$\mathcal{I}_{(b, c)}[G](t, u) = \mathcal{I}_c[\mathcal{I}_b[G]](t, u).$$

**Proof:** We prove by induction over the length  $L := |b|$ .

**Initial step:**  $L = 0$ , thus  $b = \ominus$ :

$$\mathcal{I}_c[G](t, u) = \mathcal{I}_c[\mathcal{I}_\ominus[G]](t, u).$$

**Induction step:**  $L - 1 \rightarrow L, L \geq 1$ : There exist a multi-index  $b' \in \mathcal{M}_m$  and an index  $\beta \in \{0, \dots, m\}$  with  $b = (\beta, b')$ . We apply now definition 2.4 and the induction hypothesis and get

$$\begin{aligned} \mathcal{I}_{(\beta, b', c)}[G](t, u) &= \mathcal{I}_{(b', c)} \left[ \int_{\tau}^{\cdot} G(s, u) u^{\beta}(s) \, ds \right] (t, u) \\ &= \mathcal{I}_c \left[ \mathcal{I}_{b'} \left[ \int_{\tau}^{\cdot} G(s, u) u^{\beta}(s) \, ds \right] \right] (t, u) \\ &= \mathcal{I}_c [\mathcal{I}_{(\beta, b')}[G]](t, u). \end{aligned}$$

■

## 2.3 Integral representations

The purpose of the iterated integrals we defined in the last section is the integral representation for the solution as well as for the right hand side vector fields depending on the solution. We use the notation of Grüne and Kloeden [12]. The integral representation uses the Lie-derivatives of the right hand side vector fields. We assume the existence of all used partial derivatives of the vector fields  $f^\alpha$ ,  $\alpha \in \{1, \dots, m\}$ .

### Theorem 2.12 (Integral representation for the solution)

Consider an *hierarchical* multi-index set  $\mathcal{H} \subset \mathcal{M}_m$ . Then any solution of the control system (1.1) can be represented as

$$x(t) = \sum_{b \in \mathcal{H}} \mathcal{L}^b x(\tau) \mathcal{I}_b(t, u) + \sum_{b \in \mathcal{R}(\mathcal{H})} \mathcal{I}_b \left[ \mathcal{L}^b x \right] (t, u). \quad (2.6)$$

**Proof:** We prove the assertion by induction over the cardinal number of the set  $\mathcal{H}$ . For simplicity we skip the argument  $(t, u)$  in the notation of the iterated integrals.

**Initial step**  $|\mathcal{H}| = 1 \Leftrightarrow \mathcal{H} = \{\ominus\}$ :

We deduce the assertion directly from the integral equation (2.2) of the control affine system. The remainder set  $\mathcal{R}(\mathcal{H}) = \{(0), \dots, (m)\}$ .

$$\begin{aligned}
 x(t) &= x(\tau) + \sum_{\alpha=1}^m \int_{\tau}^t f^{\alpha}(s, x(s)) u^{\alpha}(s) \, ds \\
 &= x(\tau) + \sum_{\alpha=0}^m \int_{\tau}^t \mathcal{L}^{(\alpha)} x(s) u^{\alpha}(s) \, ds \\
 &= x(\tau) + \sum_{\alpha=0}^m \mathcal{I}_{(\alpha)} \left[ \mathcal{L}^{(\alpha)} x \right] \\
 &= \mathcal{L}^{\ominus} x(\tau) \mathcal{I}_{\ominus} [1] + \sum_{b \in \mathcal{R}(\mathcal{H})} \mathcal{I}_b \left[ \mathcal{L}^b x \right]
 \end{aligned}$$

**Induction step**  $|\mathcal{H}| - 1 \rightarrow |\mathcal{H}|$ :

We construct a hierarchical set  $\mathcal{G} \subset \mathcal{H}$  with  $|\mathcal{G}| = |\mathcal{H}| - 1$ . Denote  $b_{\max} \in \mathcal{H}$  the (not necessarily unique) index with maximum length

$$|b_{\max}| \geq |b|, \quad \text{for all } b \in \mathcal{H}.$$

Then the set  $\mathcal{G} := \mathcal{H} \setminus \{b_{\max}\}$  is again hierarchical, with  $b_{\max} \in \mathcal{R}(\mathcal{G})$ . The relation of the remainder sets is

$$\mathcal{R}(\mathcal{H}) = \mathcal{R}(\mathcal{G}) \cup \{(b_{\max}, 0), \dots, (b_{\max}, m)\}.$$

Because of  $|\mathcal{G}| = |\mathcal{H}| - 1$  we can apply the induction hypothesis:

$$\begin{aligned}
 x(t) &= \sum_{b \in \mathcal{G}} \mathcal{L}^b x(\tau) \mathcal{I}_b [1] + \sum_{b \in \mathcal{R}(\mathcal{G})} \mathcal{I}_b \left[ \mathcal{L}^b x \right] \\
 &= \sum_{b \in \mathcal{G}} \mathcal{L}^b x(\tau) \mathcal{I}_b [1] + \sum_{b \in \mathcal{R}(\mathcal{G}) \setminus \{b_{\max}\}} \mathcal{I}_b \left[ \mathcal{L}^b x \right] + \mathcal{I}_{b_{\max}} \left[ \mathcal{L}^{b_{\max}} x \right].
 \end{aligned}$$



From equation (2.3) follows for the last term

$$\begin{aligned}
\mathcal{I}_{b_{\max}} \left[ \mathcal{L}^{b_{\max}} x \right] &= \\
&= \mathcal{I}_{b_{\max}} \left[ \mathcal{L}^{b_{\max}} x(\tau) + \int_{\tau}^{\cdot} \sum_{\alpha=0}^m \mathcal{L}^{(\alpha, b_{\max})} x(s) u^{\alpha}(s) \, ds \right] \\
&= \mathcal{I}_{b_{\max}} \left[ \mathcal{L}^{b_{\max}} x(\tau) \right] + \sum_{\alpha=0}^m \mathcal{I}_{b_{\max}} \left[ \int_{\tau}^{\cdot} \mathcal{L}^{(\alpha, b_{\max})} x(s) u^{\alpha}(s) \, ds \right] \\
&= \mathcal{L}^{b_{\max}} x(\tau) \mathcal{I}_{b_{\max}} [1] + \sum_{\alpha=0}^m \mathcal{I}_{(\alpha, b_{\max})} \left[ \mathcal{L}^{(\alpha, b_{\max})} x \right].
\end{aligned}$$

So we get for  $x$

$$\begin{aligned}
x(t) &= \sum_{b \in \mathcal{G}} \mathcal{L}^b x(\tau) \mathcal{I}_b [1] + \mathcal{L}^{b_{\max}} x(\tau) \mathcal{I}_{b_{\max}} [1] \\
&\quad + \sum_{b \in \mathcal{R}(\mathcal{G}) \setminus \{b_{\max}\}} \mathcal{I}_b \left[ \mathcal{L}^b x \right] + \sum_{\alpha=0}^m \mathcal{I}_{(\alpha, b_{\max})} \left[ \mathcal{L}^{(\alpha, b_{\max})} x \right] \\
&= \sum_{b \in \mathcal{H}} \mathcal{L}^b x(\tau) \mathcal{I}_b [1] + \sum_{b \in \mathcal{R}(\mathcal{H})} \mathcal{I}_b \left[ \mathcal{L}^b x \right].
\end{aligned}$$

■

Next we develop an integral representation of the right hand side vector fields  $f^{\alpha}(\cdot, x)$ ,  $\alpha \in \{1, \dots, m\}$ .

**Theorem 2.13 (integral representation of  $f^{\alpha}$ )** *Consider an arbitrary hierarchical multi-index set  $\mathcal{H} \subset \mathcal{M}_m$ . Then for every  $\alpha \in \{1, \dots, m\}$  we get an integral representation for the vector field  $f^{\alpha}$ .*

$$f^{\alpha}(t, x(t)) = \sum_{b \in \mathcal{H}} L^{(b, \alpha)} x(\tau) I_b(t, u) + \sum_{b \in \mathcal{R}(\mathcal{H})} I_b[L^{(b, \alpha)} x](t, u) \quad (2.7)$$

**Proof:** The proof has the same structure as the proof of theorem 2.12, namely the induction over the size of the hierarchical set  $\mathcal{H}$ . Again we skip the argument  $(t, u)$  of the iterated integrals.

**Initial step  $|\mathcal{H}| = 1 \Leftrightarrow \mathcal{H} = \{\ominus\}$ :**

The remainder set is  $\mathcal{R}(\mathcal{H}) = \{(0), \dots, (m)\}$ . We apply lemma 2.2 for

$f^\alpha(t, x(t))$  and use the identity  $\mathcal{L}^\alpha x = f^\alpha(\cdot, x)$ .

$$\begin{aligned}
f^\alpha(t, x(t)) &= f^\alpha(\tau, x(\tau)) + \int_\tau^t \sum_{\beta=0}^m \mathcal{L}^\beta f^\alpha(s, x(s)) u^\beta(s) \, ds \\
&= \mathcal{L}^\alpha x(\tau) + \sum_{\beta=0}^m \int_\tau^t \mathcal{L}^\beta \mathcal{L}^\alpha x(s) u^\beta(s) \, ds \\
&= \mathcal{L}^\alpha x(\tau) + \sum_{\beta=0}^m \int_\tau^t \mathcal{L}^{(\beta, \alpha)} x(s) u^\beta(s) \, ds \\
&= \mathcal{L}^\alpha x(\tau) + \sum_{b \in \mathcal{R}(\mathcal{H})} \mathcal{I}_b \left[ \mathcal{L}^{(b, \alpha)} x \right]
\end{aligned}$$

**Induction step:**  $|\mathcal{H}| - 1 \rightarrow |\mathcal{H}|$ :

We construct the hierarchical set  $\mathcal{G} \subset \mathcal{H}$  like in the proof of theorem 2.12.

Because of  $|\mathcal{G}| = |\mathcal{H}| - 1$  the induction hypothesis holds true:

$$\begin{aligned}
f^\alpha(t, x(t)) &= \sum_{b \in \mathcal{G}} \mathcal{L}^{(b, \alpha)} x(\tau) \mathcal{I}_b + \sum_{b \in \mathcal{R}(\mathcal{G})} \mathcal{I}_b \left[ \mathcal{L}^{(b, \alpha)} x \right] \\
&= \sum_{b \in \mathcal{G}} \mathcal{L}^{(b, \alpha)} x(\tau) \mathcal{I}_b \\
&\quad + \sum_{b \in \mathcal{R}(\mathcal{G}) \setminus \{b_{\max}\}} \mathcal{I}_b \left[ \mathcal{L}^{(b, \alpha)} x \right] + \mathcal{I}_{b_{\max}} \left[ \mathcal{L}^{(b_{\max}, \alpha)} x \right].
\end{aligned}$$

From equation (2.3) we get

$$\begin{aligned}
\mathcal{I}_{b_{\max}} \left[ \mathcal{L}^{(b_{\max}, \alpha)} x \right] &= \\
&= \mathcal{I}_{b_{\max}} \left[ \mathcal{L}^{(b_{\max}, \alpha)} x(\tau) + \int_\tau^t \sum_{\beta=0}^m \mathcal{L}^{(\beta, b_{\max}, \alpha)} x(s) u^\beta(s) \, ds \right] \\
&= \mathcal{I}_{b_{\max}} \left[ \mathcal{L}^{(b_{\max}, \alpha)} x(\tau) \right] + \sum_{\beta=0}^m \mathcal{I}_{b_{\max}} \left[ \int_\tau^t \mathcal{L}^{(\beta, b_{\max}, \alpha)} x(s) u^\beta(s) \, ds \right] \\
&= \mathcal{L}^{(b_{\max}, \alpha)} x(\tau) \mathcal{I}_{b_{\max}} + \sum_{\beta=0}^m \mathcal{I}_{(\beta, b_{\max})} \left[ \mathcal{L}^{(\beta, b_{\max}, \alpha)} x \right].
\end{aligned}$$

Then it follows for the vector field  $f^\alpha$

$$\begin{aligned}
 f^\alpha(t, x(t)) &= \\
 &= \sum_{b \in \mathcal{G}} \mathcal{L}^{(b, \alpha)} x(\tau) \mathcal{I}_b + \mathcal{L}^{(b_{\max}, \alpha)} x(\tau) \mathcal{I}_{b_{\max}} \\
 &\quad + \sum_{b \in \mathcal{R}(\mathcal{G}) \setminus \{b_{\max}\}} \mathcal{I}_b \left[ \mathcal{L}^{(b, \alpha)} x \right] + \sum_{\beta=0}^m \mathcal{I}_{(\beta, b_{\max})} \left[ \mathcal{L}^{(\beta, b_{\max}, \alpha)} x \right] \\
 &= \sum_{b \in \mathcal{H}} \mathcal{L}^{(b, \alpha)} x(\tau) \mathcal{I}_b + \sum_{b \in \mathcal{R}(\mathcal{H})} \mathcal{I}_b \left[ \mathcal{L}^{(b, \alpha)} x \right].
 \end{aligned}$$

■

We can choose for a given order  $p \in \mathbb{N}_0$  the hierarchical set  $\mathcal{H} := \{b \in \mathcal{M}_m \mid |b| \leq p\}$  with its remainder set  $\mathcal{R} = \mathcal{M}_m^{p+1}$  and get with the equations (2.6) and (2.7):

$$x(t) = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \mathcal{L}^b x(\tau) \mathcal{I}_b(t, u) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b \left[ \mathcal{L}^b x \right] (t, u), \quad (2.8)$$

$$f^\alpha(t, x(t)) = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \mathcal{L}^{(b, \alpha)} x(\tau) \mathcal{I}_b(t, u) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b \left[ \mathcal{L}^{(b, \alpha)} x \right] (t, u). \quad (2.9)$$

These two representations of functions depending on  $u$  are the motivation for the definition of Fliess-expansions in the next section. But first we want compute the integral representation of order  $p = 5$  for the solution of a one-dimensional example.

**Example 2.14** *We look at a simple one-dimensional control system:*

$$\dot{x} = u - x^2, \quad x(0) = 0.$$

*Using the notation from definition 1.1 we get a control affine system with  $n = 1$ ,  $m = 2$ . For the vector fields we get  $f^1(t, x) = -x^2$  and  $f^2(t, x) = 1$  and the control functions are  $u^1 \equiv 1$  and  $u^2 = u$ .*

Because  $f^1$  and  $f^2$  are autonomous, their time derivatives vanish. Thus the following implication holds true for all  $b = (\beta_{|b|}, \dots, \beta_1) \in \mathcal{M}_m$ :

$$\exists j \in \{1, \dots, |b|\} \text{ with } \beta_j = 0 \implies \mathcal{L}^b x \equiv 0.$$

$b$	$\mathcal{L}^b x$	$b$	$\mathcal{L}^b x$
$\ominus$	$x$	$(1, 1, 1, 1, 1, 1)$	$720x^7$
$(1)$	$-x^2$	$(1, 1, 1, 1, 2, 1)$	$-48x^5$
$(2)$	$1$	$(1, 1, 1, 2, 1, 1)$	$-144x^5$
$(1, 1)$	$2x^3$	$(1, 1, 2, 1, 1, 1)$	$-288x^5$
$(2, 1)$	$-2x$	$(1, 1, 2, 1, 2, 1)$	$8x^3$
$(1, 1, 1)$	$-6x^4$	$(1, 1, 2, 2, 1, 1)$	$24x^3$
$(1, 2, 1)$	$2x^2$	$(1, 2, 1, 1, 1, 1)$	$-480x^5$
$(2, 1, 1)$	$6x^2$	$(1, 2, 1, 1, 2, 1)$	$24x^3$
$(2, 2, 1)$	$-2$	$(1, 2, 1, 2, 1, 1)$	$72x^3$
$(1, 1, 1, 1)$	$24x^5$	$(1, 2, 2, 1, 1, 1)$	$144x^3$
$(1, 1, 2, 1)$	$-4x^3$	$(2, 1, 1, 1, 1, 1)$	$-720x^5$
$(1, 2, 1, 1)$	$-12x^3$	$(2, 1, 1, 1, 2, 1)$	$48x^3$
$(2, 1, 1, 1)$	$-24x^3$	$(2, 1, 1, 2, 1, 1)$	$144x^3$
$(2, 1, 2, 1)$	$4x$	$(2, 1, 2, 1, 1, 1)$	$288x^3$
$(2, 2, 1, 1)$	$12x$	$(2, 1, 2, 1, 2, 1)$	$-8x$
$(1, 1, 1, 1, 1)$	$-120x^6$	$(2, 1, 2, 2, 1, 1)$	$-24x$
$(1, 1, 1, 2, 1)$	$12x^4$	$(2, 2, 1, 1, 1, 1)$	$480x^3$
$(1, 1, 2, 1, 1)$	$36x^4$	$(2, 2, 1, 1, 2, 1)$	$-24x$
$(1, 2, 1, 1, 1)$	$72x^4$	$(2, 2, 1, 2, 1, 1)$	$-72x$
$(1, 2, 1, 2, 1)$	$-4x^2$	$(2, 2, 2, 1, 1, 1)$	$-144x$
$(1, 2, 2, 1, 1)$	$-12x^2$		
$(2, 1, 1, 1, 1)$	$120x^4$		
$(2, 1, 1, 2, 1)$	$-12x^2$		
$(2, 1, 2, 1, 1)$	$-36x^2$		
$(2, 2, 1, 1, 1)$	$-72x^2$		
$(2, 2, 1, 2, 1)$	$4$		
$(2, 2, 2, 1, 1)$	$12$		

Table 2.1: The non-vanishing Lie derivatives up to order 6

We compute for all multi-indices  $b \in \mathcal{M}_m$  with  $|b| \leq p+1 = 6$  the Lie derivatives  $\mathcal{L}^b x$ . The non vanishing ones are itemised in table 2.1. We insert the initial condition  $x(0) = 0$  and get only four non zero coefficients (blue items in table 2.1). Equation (2.8) becomes

$$\begin{aligned} \lambda(t, u) = & \mathcal{I}_{(2)}(t, u) - 2 \mathcal{I}_{(2, 2, 1)}(t, u) + 4 \mathcal{I}_{(2, 2, 1, 2, 1)}(t, u) \\ & + 12 \mathcal{I}_{(2, 2, 2, 1, 1)}(t, u) + R_\lambda^6(t, u), \end{aligned} \quad (2.10)$$

where the remainder term is:

$$\begin{aligned} R_\lambda^6 = & \sum_{b \in \mathcal{M}_m^6} \mathcal{I}_b [L^b x] = \\ = & \mathcal{I}_{(1, 1, 1, 1, 1, 1)} [720x^7] + \mathcal{I}_{(1, 1, 1, 1, 2, 1)} [-48x^5] + \\ & + \mathcal{I}_{(1, 1, 1, 2, 1, 1)} [-144x^5] + \mathcal{I}_{(1, 1, 2, 1, 1, 1)} [-288x^5] + \\ & + \mathcal{I}_{(1, 1, 2, 1, 2, 1)} [8x^3] + \mathcal{I}_{(1, 1, 2, 2, 1, 1)} [24x^3] + \\ & + \mathcal{I}_{(1, 2, 1, 1, 1, 1)} [-480x^5] + \mathcal{I}_{(1, 2, 1, 1, 2, 1)} [24x^3] + \\ & + \mathcal{I}_{(1, 2, 1, 2, 1, 1)} [72x^3] + \mathcal{I}_{(1, 2, 2, 1, 1, 1)} [144x^3] + \\ & + \mathcal{I}_{(2, 1, 1, 1, 1, 1)} [-720x^5] + \mathcal{I}_{(2, 1, 1, 1, 2, 1)} [48x^3] + \\ & + \mathcal{I}_{(2, 1, 1, 2, 1, 1)} [144x^3] + \mathcal{I}_{(2, 1, 2, 1, 1, 1)} [288x^3] + \\ & + \mathcal{I}_{(2, 1, 2, 1, 2, 1)} [-8x] + \mathcal{I}_{(2, 1, 2, 2, 1, 1)} [-24x] + \\ & + \mathcal{I}_{(2, 2, 1, 1, 1, 1)} [480x^3] + \mathcal{I}_{(2, 2, 1, 1, 2, 1)} [-24x] + \\ & + \mathcal{I}_{(2, 2, 1, 2, 1, 1)} [-72x] + \mathcal{I}_{(2, 2, 2, 1, 1, 1)} [-144x]. \end{aligned}$$

We skipped the argument  $(t, u)$  for clearness and denote with  $\lambda(t, u) := \lambda(t, \tau, x_0, u)$  the solution for  $\tau = 0$  and  $x_0 = 0$ .

## 2.4 Fliess-expansions

The following definition of causality continues along the definition of Hinrichsen and Pritchard [14, definition 2.3.24]. It is a very natural condition requiring that the value of a control dependent function, only depends on the past part but not on the future part of the control functions.

**Definition 2.15 (causality condition)** *Consider  $G : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  as a function which depends on the time and on the vector of control functions. It fulfils the causality condition if for every  $s \in [\tau, T]$  and for all  $u, v \in \mathcal{U}^m$  with  $u|_{[\tau, s]} = v|_{[\tau, s]}$  the following equation is satisfied:*

$$G(\cdot, u)|_{[\tau, s]} = G(\cdot, v)|_{[\tau, s]}$$

*We call a function, which fulfils the causality condition, causal function.*

Chen and Fliess introduced the Fliess-expansion (see [7, 8]), which is the series expansion for the solutions of control affine systems. They were interested in the convergence if the order tends to infinity and in the uniqueness. Convergence is assured by constraints on the control functions and on the right hand side vector fields.

**Definition 2.16 (Fliess-expansion)** *For a given order  $p \in \mathbb{N}$  we define a Fliess-expansion of a function  $\mu : [\tau, T] \times \mathcal{U}^m \rightarrow \mathbb{R}^n$  by*

$$\mu(t, u) = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (\mu)_b \mathcal{I}_b(t, u) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[\mu_b](t, u),$$

where  $u := (u^1, \dots, u^m)$  is the vector of control functions. The coefficients  $(\mu)_b \in \mathbb{R}^n$ , for  $|b| \leq p$  are constant vectors and the integral kernels  $\mu_b : [\tau, T] \times \mathcal{U}^m \rightarrow \mathbb{R}^n$ , for  $|b| = p + 1$  are causal functions.

The integral representation of the solution in equation (2.8) is a Fliess-expansion of order  $p$ , because the Lie-derivatives in the remainder term are causal. Thus the following definition is well-defined.

**Definition 2.17 (Fliess-expansion of the general solution)**

*We call the Fliess-expansion of order  $p$  of the general solution 1.2:*

$$\begin{aligned} \lambda(t, \tau, x_0, u) &= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (\lambda(\tau, x_0))_b \mathcal{I}_b(t, u) + R_{\lambda(\tau, x_0)}^{p+1}(t, u), \\ \text{with } (\lambda(\tau, x_0))_b &:= \mathcal{L}^b x(\tau, x_0), \\ R_{\lambda(\tau, x_0)}^{p+1}(t, u) &:= \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[\mathcal{L}^b x(\tau, x_0)](t, u), \end{aligned}$$

where  $u := (u^1, \dots, u^m)^T$  denotes the vector of control functions and  $x(\tau) = x_0$  gives the initial condition.

We skipped the vector of control functions  $u$  in the denotation of the coefficients  $(\lambda(\tau, x_0))_b$ . Because of the causality of the Lie-derivatives of the solution the coefficients do not depend on  $u$ .

**Remark 2.18 (Volterra-expansions)** *There is another series expansion for the solution of control systems. It is known as the Volterra-expansion (cp. Isidori [17, Chapter 3.2]). It is defined in a more general context than*

the Fliess-expansion and has non-constant integral kernels for the coefficients. The uniqueness of Volterra-expansions is discussed by Lesiak and Krener [25].

## 2.5 Uniqueness of Fliess-expansions

Important for us is the uniqueness of the coefficients of finite Fliess-expansions. Fliess has already given an algebraically motivated proof for the uniqueness of Fliess-expansions in [9]. Our proof will only use methods from analysis to fit better in the framework of this work. There is also an older proof of Fliess in [7] which is cited by Isidori [17]. It turned out that this version is rather incomplete.

Now we define a very technical tool. We construct based on a given vector of control functions the vector of control functions chopped off at a fixed time.

### Definition 2.19 (chopped-off control functions)

Based on the given vector of control functions  $u = (u^1, \dots, u^m) \in \mathcal{U}^m$ , a time  $s \in [\tau, T]$  and a coordinate  $\alpha \in \{0, \dots, m\}$  we construct a new control vector  ${}_s u_\alpha := ({}_s u_\alpha^1, \dots, {}_s u_\alpha^m) \in \mathcal{U}^m$ . It coincides up to time  $s$  with the vector  $u$  and afterwards with the  $\alpha$ -unit vector (Thereby the 0-unit vector is assumed to be the zero vector.):

$${}_s u_\alpha^\beta := \begin{cases} u^\beta, & \text{on } [\tau, s) \\ 1, & \text{on } [s, T], \text{ for } \beta = \alpha \\ 0, & \text{on } [s, T], \text{ for } \beta \neq \alpha \end{cases}$$

Again by convention we define  ${}_s u_\alpha^0 \equiv 1$ .

Here the concept of causality comes into play. In particular a causal  $G : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  satisfies due to  ${}_s u_\alpha|_{[\tau, s)} = u|_{[\tau, s)}$  the equation

$$G(s, {}_s u_\alpha) = G(s, {}_s u_\alpha)|_{[\tau, s]} = G(s, u)|_{[\tau, s]} = G(s, u), \quad (2.11)$$

for all  $\alpha \in \{0, \dots, m\}$ .

**Lemma 2.20** *Let the function  $G : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  fulfil the causality condition. Then for every multi-index  $b \in \mathcal{M}_m$  the iterated integral with integral kernel  $G$  satisfies the equation*

$$\mathcal{I}_b[G](s, {}_s u_\alpha) = \mathcal{I}_b[G](s, u),$$

for all  $\alpha \in \{0, \dots, m\}$ .

**Proof:** We prove by induction over the length  $|b|$ .

**Initial step**  $|b| = 0$ , **thus**  $b = \ominus$ : With equation 2.11 we get

$$\mathcal{I}_{\ominus}[G](s, {}_s u_{\alpha}) = G(s, {}_s u_{\alpha}) = G(s, u) = \mathcal{I}_{\ominus}[G](s, u).$$

**Induction step**  $|b| \rightarrow |(b, \beta)|, \beta \in \{0, \dots, m\}$ : It follows from the induction hypothesis, from  ${}_s u_{\alpha}^{\beta}(s_1) = u^{\beta}(s_1)$ , for  $s_1 < s$ , and from lemma 2.7:

$$\begin{aligned} \mathcal{I}_{(b, \beta)}[G](s, {}_s u_{\alpha}) &= \int_{\tau}^s \mathcal{I}_b[G](s_1, {}_s u_{\alpha}) {}_s u_{\alpha}^{\beta}(s_1) ds_1 \\ &= \int_{\tau}^s \mathcal{I}_b[G](s_1, u) u^{\beta}(s_1) ds_1 \\ &= \mathcal{I}_{(b, \beta)}[G](s, u). \end{aligned}$$

■

**Lemma 2.21** *Let the function  $G : [\tau, T] \times \mathcal{U}^m \rightarrow \mathbb{R}^n$  be a integrable function. Then the iterated integral for any multi-index  $b \in \mathcal{M}_m$ , except the empty index, vanishes at the initial time  $\tau$ :*

$$\mathcal{I}_b[G](\tau, u) = \begin{cases} G(\tau, u), & \text{for } b = \ominus, \\ 0, & \text{otherwise.} \end{cases}$$

**Proof:** First we prove the trivial case, where  $b$  is the empty index. Then we will look at the nonempty indices.

- $b = \ominus$ :

$$\mathcal{I}_{\ominus}[G](\tau, u) = G(\tau, u)$$

- $|b| > 0$ , thus there exists a multi-index  $b' \in \mathcal{M}_m$  and a number  $\beta \in \{0, \dots, m\}$ , with  $b = (b', \beta)$ :

$$\mathcal{I}_b[G](\tau, u) = \int_{\tau}^{\tau} \mathcal{I}_{b'}[G](s_1, u) u^{\beta}(s_1) ds_1 = 0$$

Here we get a non-degenerated iterated integral from  $\tau$  to  $\tau$ .

■

Before we can prove the uniqueness of Fliess-expansions we need to develop a tool to isolate an arbitrary coefficient. Then we can compare two expansions coefficient-wise. Therefore we utilise the previously defined chopped-off functions (see definition 2.19).



**Definition 2.22 (Fliess-derivatives)** Let  $\mu : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  be a function which depends on the vector of control functions  $u = (u^1, \dots, u^m) \in \mathcal{U}^m$ . For  $\alpha \in \{0, \dots, m\}$  we define

$$\begin{aligned} \alpha = 0 : \quad \mathcal{F}_0[\mu](s, u) &:= \lim_{h \rightarrow 0} \frac{\mu(s+h, {}_s u_0) - \mu(s, {}_s u_0)}{h} \\ \alpha = 1, \dots, m : \quad \mathcal{F}_\alpha[\mu](s, u) &:= \lim_{h \rightarrow 0} \frac{\mu(s+h, {}_s u_\alpha) - \mu(s, {}_s u_\alpha)}{h} - \mathcal{F}_0[\mu](s, u). \end{aligned}$$

The concatenation for  $b = (\beta_{|b|}, \dots, \beta_1) \in \mathcal{M}_m$  is denoted by

$$\mathcal{F}_b[\mu](s, u) := \mathcal{F}_{\beta_{|b|}} \cdots \mathcal{F}_{\beta_1}[\mu](s, u).$$

**Remark 2.23** The operator  $\mathcal{F}_\alpha$  again creates a function, which depends on the time and the vector of control functions and has values in  $\mathbb{R}^n$ :

$$\mathcal{F}_\alpha[\mu] : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n.$$

Hence the concatenation  $\mathcal{F}_b$  is well defined.

Now we analyse some properties of the Fliess-derivative. Later we can detect its effect on Fliess-expansions.

**Lemma 2.24 (linearity of Fliess-derivatives)**

For all  $b \in \mathcal{M}_m$  the Fliess-derivative  $\mathcal{F}_b$  is linear.

**Proof:** Let  $\mu, \nu : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  be functions and let  $n_1, n_2 \in \mathbb{R}$  be arbitrary real numbers. Then we can show the linearity of the operators  $\mathcal{F}_0$  and  $\mathcal{F}_\alpha$ , for  $\alpha = 1, \dots, m$ :

$$\begin{aligned} \mathcal{F}_0[n_1\mu + n_2\nu](s, u) &= \\ &= \lim_{h \rightarrow 0} \frac{n_1\mu(s+h, {}_s u_0) + n_2\nu(s+h, {}_s u_0)}{h} - \frac{n_1\mu(s, {}_s u_0) + n_2\nu(s, {}_s u_0)}{h} \\ &= \lim_{h \rightarrow 0} \frac{n_1\mu(s+h, {}_s u_0) - n_1\mu(s, {}_s u_0)}{h} + \lim_{h \rightarrow 0} \frac{n_2\nu(s+h, {}_s u_0) - n_2\nu(s, {}_s u_0)}{h} \\ &= n_1\mathcal{F}_0[\mu](s, u) + n_2\mathcal{F}_0[\nu](s, u), \end{aligned}$$

and

$$\begin{aligned}
\mathcal{F}_\alpha[n_1\mu + n_2\nu](s, u) &= \\
&= \lim_{h \rightarrow 0} \frac{n_1\mu(s+h, {}_s u_\alpha) + n_2\nu(s+h, {}_s u_\alpha)}{h} - \frac{n_1\mu(s, {}_s u_\alpha) + n_2\nu(s, {}_s u_\alpha)}{h} - \\
&\quad - \mathcal{F}_0[n_1\mu + n_2\nu](s) \\
&= \lim_{h \rightarrow 0} \frac{n_1\mu(s+h, {}_s u_\alpha) - n_1\mu(s, {}_s u_\alpha)}{h} - n_1\mathcal{F}_0[\mu](s) \\
&\quad + \lim_{h \rightarrow 0} \frac{n_2\nu(s+h, {}_s u_\alpha) - n_2\nu(s, {}_s u_\alpha)}{h} - n_2\mathcal{F}_0[\nu](s) \\
&= n_1\mathcal{F}_\alpha[\mu](s, u) + n_2\mathcal{F}_\alpha[\nu](s, u).
\end{aligned}$$

Of course, the linearity carries forward to the concatenation of Fliess-derivatives  $\mathcal{F}_b$ , for all  $b \in \mathcal{M}_m$ . ■

For control independent functions the Fliess-derivative coincides for all multi-indices consisting only of zeros with the conventional time-derivative. Otherwise it vanishes.

**Lemma 2.25** *Let  $G : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  be a function, which is independent of the vector of control functions and is sufficiently many times differentiable in the first component, thus  $G(s, u) = G(s)$ . Then the following equation holds true for all  $b \in \mathcal{M}_m$ :*

$$\mathcal{F}_b[G](s, u) = \begin{cases} G^{(|b|)}(s), & \text{for } b = (0, \dots, 0), \\ 0, & \text{otherwise.} \end{cases}$$

Thereby  $G^{(k)}$  denotes the  $k$ -th derivative concerning the first component.

**Proof:** First we prove the assertion for  $|b| = 1$ ,

$$\mathcal{F}_0[G](s, u) = \lim_{h \rightarrow 0} \frac{G(s+h) - G(s)}{h} = G'(s).$$

For  $\alpha = 1, \dots, m$  it applies:

$$\mathcal{F}_\alpha[G](s, u) = \lim_{h \rightarrow 0} \frac{G(s+h) - G(s)}{h} - \mathcal{F}_0[G](s, u) = 0.$$

For multi-indices of length  $i \in \{2, 3, \dots\}$  follows after  $i$ -times recurrence immediately

$$\mathcal{F}_{(\beta_i, \dots, \beta_1)}[G](s, u) = \begin{cases} G^{(i)}(s), & \text{for } \beta_1 = \dots = \beta_i = 0, \\ 0, & \text{otherwise.} \end{cases}$$

■

Now we analyse the effect of Fliess-derivatives on iterated integrals in order to apply them later to Fliess-expansions.

**Lemma 2.26 (Fliess-derivative for iterated integrals)**

Let  $G : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  be a function, which is sufficiently many times differentiable in the first component and fulfils the causality condition. Then the Fliess-derivative of an iterated integral is

$$\mathcal{F}_\alpha [\mathcal{I}_{(b, \beta)} [G]] (s, u) = \begin{cases} \mathcal{I}_b [G] (s, u), & \text{for } \beta = \alpha, \\ 0, & \text{otherwise,} \end{cases}$$

$$\mathcal{F}_a [\mathcal{I}_b [G]] (s, u) = \begin{cases} \mathcal{I}_c [G] (s, u), & \text{for } b = (c, a), \\ \mathcal{F}_c [G] (s, u), & \text{for } a = (c, b), \\ 0, & \text{otherwise.} \end{cases}$$

Thereby  $a, b \in \mathcal{M}_m$  are two arbitrary multi-indices and  $\alpha, \beta \in \{0, \dots, m\}$ .

In the second equation the conditions are abbreviated. With this formulations we mean, that there exists a multi-index  $c \in \mathcal{M}_m$  with  $b = (c, a)$  and  $a = (c, b)$ , respectively.

**Proof:** For the case  $\alpha = 0$  of the first assertion we apply lemma 2.7 to the definition 2.22 of Fliess-derivatives:

$$\begin{aligned} \mathcal{F}_0 [\mathcal{I}_{(b, \beta)} [G]] (s, u) &= \lim_{h \rightarrow 0} \frac{\mathcal{I}_{(b, \beta)} [G] (s + h, {}_s u_0) - \mathcal{I}_{(b, \beta)} [G] (s, {}_s u_0)}{h} \\ &= \lim_{h \rightarrow 0} \frac{1}{h} \left( \int_s^{s+h} \mathcal{I}_b [G] (s_1, {}_s u_0) {}_s u_0^\beta (s_1) ds_1 \right). \end{aligned}$$

For  $s_1 > s$  is  ${}_s u_0^\beta (s_1) = 1$  for  $\beta = 0$  and it vanishes for  $\beta > 0$ , thus

$$\begin{aligned} \mathcal{F}_0 [\mathcal{I}_{(b, 0)} [G]] (s, u) &= \lim_{h \rightarrow 0} \frac{1}{h} \left( \int_s^{s+h} \mathcal{I}_b [G] (s_1, {}_s u_0) ds_1 \right) \\ &= \mathcal{I}_b [G] (s, {}_s u_0) \\ &= \mathcal{I}_b [G] (s, u). \end{aligned}$$

The last step follows from lemma 2.20. Here the causality concept applies. We combine to

$$\mathcal{F}_0 [\mathcal{I}_{(b, \beta)} [G]] (s, u) = \begin{cases} \mathcal{I}_b [G] (s, u), & \text{for } \beta = 0, \\ 0, & \text{otherwise.} \end{cases}$$

Now we prove the first assertion for  $\alpha \in \{1, \dots, m\}$ :

$$\begin{aligned} \mathcal{F}_\alpha [\mathcal{I}_{(b, \beta)} [G]](s, u) &= \lim_{h \rightarrow 0} \frac{\mathcal{I}_{(b, \beta)} [G](s + h, {}_s u_\alpha) - \mathcal{I}_{(b, \beta)} [G](s, {}_s u_\alpha)}{h} \\ &\quad - \mathcal{F}_0 [\mathcal{I}_{(b, \beta)} [G]](s, u) \\ &= \lim_{h \rightarrow 0} \frac{1}{h} \left( \int_s^{s+h} \mathcal{I}_b [G](s_1, {}_s u_\alpha) {}_s u_\alpha^\beta(s_1) ds_1 \right) \\ &\quad - \mathcal{F}_0 [\mathcal{I}_{(b, \beta)} [G]](s, u). \end{aligned}$$

For  $0 = \beta \neq \alpha$  follows, where  ${}_s u_\alpha^0 \equiv 1$  by convention:

$$\begin{aligned} \mathcal{F}_\alpha [\mathcal{I}_{(b, 0)} [G]](s, u) &= \lim_{h \rightarrow 0} \frac{1}{h} \left( \int_s^{s+h} \mathcal{I}_b [G](s_1, {}_s u_\alpha) {}_s u_\alpha^0(s_1) ds_1 \right) \\ &\quad - \mathcal{F}_0 [\mathcal{I}_{(b, 0)} [G]](s, u) \\ &= \mathcal{I}_b [G](s, {}_s u_\alpha) - \mathcal{I}_b [G](s, u) = 0, \end{aligned}$$

which follows again from lemma 2.20 with causality. For  $0 \neq \beta = \alpha$  we get, remembering  ${}_s u_\alpha^\alpha \equiv 1$  on  $[s, T]$ :

$$\begin{aligned} \mathcal{F}_\alpha [\mathcal{I}_{(b, \alpha)} [G]](s, u) &= \lim_{h \rightarrow 0} \frac{1}{h} \left( \int_s^{s+h} \mathcal{I}_b [G](s_1, {}_s u_\alpha) {}_s u_\alpha^\alpha(s_1) ds_1 \right) \\ &\quad - \mathcal{F}_0 [\mathcal{I}_{(b, \alpha)} [G]](s, u) \\ &= \mathcal{I}_b [G](s, {}_s u_\alpha) \\ &= \mathcal{I}_b [G](s, u). \end{aligned}$$

And for  $0 \neq \beta \neq \alpha$  follows ( ${}_s u_\alpha^\beta \equiv 0$  on  $[s, T]$ ):

$$\begin{aligned} \mathcal{F}_\alpha [\mathcal{I}_{(b, \beta)} [G]](s, u) &= \lim_{h \rightarrow 0} \frac{1}{h} \left( \int_s^{s+h} \mathcal{I}_b [G](s_1, {}_s u_\alpha) {}_s u_\alpha^\beta(s_1) ds_1 \right) \\ &\quad - \mathcal{F}_0 [\mathcal{I}_{(b, \beta)} [G]](s, u) \\ &= 0. \end{aligned}$$

After collecting all cases the first assertion is proved:

$$\begin{aligned} \mathcal{F}_\alpha [\mathcal{I}_{(b, \beta)} [G]](s, u) &= \lim_{h \rightarrow 0} \frac{1}{h} (\mathcal{I}_{(b, \beta)} [G](s + h, u) - \mathcal{I}_{(b, \beta)} [G](s, u)) \\ &= \begin{cases} \mathcal{I}_b [G](s), & \text{for } \beta = \alpha, \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

Now we check the second assertion, which describes the multiple Fliess-derivatives:

$$\mathcal{F}_a[\mathcal{I}_b[G]](s, u) = \begin{cases} \mathcal{I}_c[G](s, u), & \text{for } b = (c, a), \\ \mathcal{F}_c[G](s, u), & \text{for } a = (c, b), \\ 0, & \text{otherwise.} \end{cases}$$

For the trivial case  $b = \ominus$  we get

$$\mathcal{F}_a[\mathcal{I}_\ominus[G]](s, u) = \mathcal{F}_a[G](s, u),$$

which fulfils the second case of the assertion. For the case  $b \neq \ominus$  there exist  $b' \in \mathcal{M}_m$  and  $\beta \in \{0, \dots, m\}$  with  $b = (b', \beta)$ . Now we can give the proof by induction over the length  $L := |a|$ .

**Initial step:**  $L = 0$ , **thus**  $a = \ominus$ : In this trivial case we apply the Fliess-derivative with empty index, which is the identity by definition:

$$\mathcal{F}_\ominus[\mathcal{I}_b[G]](s, u) = \mathcal{I}_b[G](s, u).$$

This comes up to the first case of the assertion.

**Induction step:**  $L - 1 \rightarrow L$ : Thus there is a multi-index  $a' \in \mathcal{M}$  and an index  $\alpha \in \{0, \dots, m\}$  with  $a = (a', \alpha)$ . Then we get from the first assertion

$$\begin{aligned} \mathcal{F}_a[\mathcal{I}_b[G]](s, u) &= \mathcal{F}_{a'}\mathcal{F}_\alpha[\mathcal{I}_{(b', \beta)}[G]](s, u) \\ &= \begin{cases} \mathcal{F}_{a'}[\mathcal{I}_{b'}[G]](s, u), & \text{for } \beta = \alpha, \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

Because all the others vanish, we only need to deal with the case  $\beta = \alpha$ . Therefore it follows from the induction hypothesis:

$$\mathcal{F}_a[\mathcal{I}_b[G]](s, u) = \mathcal{F}_{a'}[\mathcal{I}_{b'}[G]](s, u) = \begin{cases} \mathcal{I}_c[G](s, u), & \text{for } b' = (c, a'), \\ \mathcal{F}_c[G](s, u), & \text{for } a' = (c, b'), \\ 0, & \text{otherwise.} \end{cases}$$

Last we transform the conditions on the multi-indices to get the conditions for the extended indices  $(a, \alpha)$  and  $(b, \beta)$ :

$$\begin{aligned} b' = (c, a') &\iff (b', \beta) = (c, a', \alpha) \iff b = (c, a), \\ a' = (c, b') &\iff (a', \alpha) = (c, b', \beta) \iff a = (c, b). \end{aligned}$$

For the case  $\beta \neq \alpha$  the derivative of the iterated integral  $\mathcal{F}_a[\mathcal{I}_b[G]]$  vanishes and becomes part of the “otherwise”-case of the second assertion.  $\blacksquare$

Next we evaluate the Fliess-derivative of iterated integrals at the initial time  $\tau$  and get the simple result, that it gives only a nonzero value if the multi-indices of the derivative and the integral coincide.

**Lemma 2.27 (Fliess-derivative for iterated integrals at  $\tau$ )** *For arbitrary multi-indices  $a, b \in \mathcal{M}_m$  we get:*

$$\mathcal{F}_a[\mathcal{I}_b](\tau, u) = \begin{cases} 1, & \text{for } a = b, \\ 0, & \text{otherwise.} \end{cases}$$

For  $|a| < |b|$  follows for arbitrary causal integral kernels  $G$

$$\mathcal{F}_a[\mathcal{I}_b[G]](\tau, u) = 0.$$

**Proof:** For  $G \equiv 1$  lemma 2.26 becomes

$$\mathcal{F}_a[\mathcal{I}_b](s, u) = \begin{cases} \mathcal{I}_c(s, u), & \text{for } b = (c, a), \\ \mathcal{F}_c[1](s, u), & \text{for } a = (c, b), \\ 0, & \text{otherwise.} \end{cases}$$

In particular follows with  $s = \tau$ :

$$\mathcal{I}_c(\tau, u) = \begin{cases} 1, & \text{for } c = \ominus, \\ 0, & \text{otherwise,} \end{cases} \quad \text{and } \mathcal{F}_c[1](\tau, u) = \begin{cases} 1, & \text{for } c = \ominus, \\ 0, & \text{otherwise.} \end{cases}$$

The conditions  $c = \ominus$  and  $a = b$  are equivalent. That implies the first assertion. For the second assertion with  $|a| < |b|$  we use again lemma 2.26

$$\mathcal{F}_a[\mathcal{I}_b[G]](\tau, u) = \begin{cases} \mathcal{I}_c[G](\tau, u), & \text{for } b = (c, a), \\ 0, & \text{otherwise,} \end{cases}$$

where  $|c| > 0$ . Lemma 2.21 yields to

$$\mathcal{F}_a[\mathcal{I}_b[G]](\tau, u) = 0.$$

■

We defined the Fliess-derivative and discussed its effect on iterated integrals and Fliess-expansions. This allows us now to prove the uniqueness of the coefficients of finite Fliess-expansions.

**Theorem 2.28 (uniqueness of Fliess-expansions)**

Let  $y, z : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  be functions with a given Fliess-expansion of order  $p$  and denote their coefficients by  $(y)_b$  and  $(z)_b$ , respectively, for  $|b| \leq p$ . Then the following implication holds:

$$y(s, u) = z(s, u) \quad \forall_{s \in [\tau, T], u \in \mathcal{U}^m} \implies (y)_b = (z)_b \quad \forall_{b \in \mathcal{M}_m, |b| \leq p}.$$

**Proof:** We define the new function  $\mu(t, u) := y(t, u) - z(t, u) \equiv 0$ . There exist causal functions  $y_b$  and  $z_b$  for all  $b \in \mathcal{M}_m^{p+1}$  as kernels of the remainder integrals of  $y$  and  $z$ . We insert the Fliess-expansion into the definition of  $\mu$ :

$$\begin{aligned} \mu(s, u) &= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (y)_b \mathcal{I}_b(s, u) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[y_b](s, u) \\ &\quad - \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (z)_b \mathcal{I}_b(s, u) - \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[z_b](s, u). \end{aligned}$$

By applying lemma 2.10 we subtract the iterated integrals and get:

$$\mu(s, u) = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} ((y)_b - (z)_b) \mathcal{I}_b(s, u) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[y_b - z_b](s, u).$$

Thereby the functions  $y_b - z_b$ , for all  $b \in \mathcal{M}_m^{p+1}$ , are causal. For every  $c \in \mathcal{M}_m$  with  $|c| \leq p$  we apply the Fliess-derivative to  $\mu$  and evaluate at  $s = \tau$ . We recall the linearity of the operator  $\mathcal{F}_c$  (lemma 2.24).

$$\begin{aligned} \mathcal{F}_c[\mu](\tau, u) &= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} ((y)_b - (z)_b) \mathcal{F}_c[\mathcal{I}_b](\tau, u) \\ &\quad + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{F}_c[\mathcal{I}_b[y_b - z_b]](\tau, u). \end{aligned}$$

With Lemma 2.27 we get

$$\mathcal{F}_c[\mu](\tau, u) = (y)_c - (z)_c.$$

From linearity of  $\mathcal{F}_c$  (lemma 2.24) and from the assumption  $\mu \equiv 0$  we know

$$\mathcal{F}_c[\mu](\tau, u) = 0,$$

what is equivalent to  $(y)_c = (z)_c$ . ■

The uniqueness of Fliess-expansions is one of basic tools in this work. It allows us to compare the Fliess-expansion for the derivative of the solution of the control system with the right hand side computed out of the solution's Fliess-expansion coefficient-wise. This will give us the possibility to rewrite the system as conditions on the coefficients.

## 2.6 Operations on Fliess-expansions

For solving problems we are interested in computing Fliess-expansions for functions of Fliess-expansions. We consider a function  $f : [\tau, T] \times \mathbb{R}^n \longrightarrow \mathbb{R}^n$  and a given Fliess-expansion  $x$  of order  $p \in \mathbb{N}$ . Then we assume the existence of the Fliess-expansion of  $f(\cdot, x)$  of order  $p$ .

Therefore we need to know how  $f$  is looking like. If for instance  $f$  is a composition of basic operations on known Fliess-expansions, we need to define the basic operations for Fliess-expansions.

In this section we will give the Fliess-expansion for some basic functions like constants or only time dependent sufficiently often differentiable functions. Then we will learn how to perform the basic operations addition, multiplication and division on Fliess-expansions. If the Taylor-expansion of order  $p$  for the function  $f$  is known we will learn how to compute the Fliess-expansion for  $f(\cdot, x)$  and prove its existence.

### 2.6.1 Special Fliess-expansions

To assemble  $f$  out of basic operations it is important to know the Fliess-expansions of “simple functions”. Without proof we point out the existence of Fliess-expansions for the following expressions. The constant  $\mu \in \mathbb{R}^n$  has the Fliess-coefficients

$$(\mu)_b = \begin{cases} \mu, & b = \ominus, \\ 0, & \text{otherwise,} \end{cases}$$

for all  $b \in \mathcal{M}_m$ ,  $|b| \leq p$ . Its remainder vanishes. The time  $t \in [\tau, T]$  has the Fliess-coefficients

$$(t)_b = \begin{cases} \tau, & b = \ominus, \\ 1, & b = (0), \\ 0, & \text{otherwise,} \end{cases}$$

for all  $b \in \mathcal{M}_m$ ,  $|b| \leq p$ , which can be easily proved by writing down the Fliess-expansion for an arbitrary order  $p \in \mathbb{N}$ . Of course, its remainder



vanishes, too. For a given Taylor-expansion of a  $(p+1)$ -times differentiable function  $g : [\tau, T] \longrightarrow \mathbb{R}^n$  we get the corresponding Fliess-expansion by

$$(g(t))_b = \begin{cases} \frac{g^{(|b|)}(\tau)}{i!}, & b = (0, \dots, 0), \\ 0, & \text{otherwise,} \end{cases}$$

for all  $b \in \mathcal{M}_m$ ,  $|b| \leq p$ , which can again be easily proved by writing down the Fliess-expansion for an arbitrary order  $p \in \mathbb{N}$ . As remainder term we get

$$\mathcal{I}_{(\underbrace{0, \dots, 0}_{|\cdot|=p+1})} [g^{(p+1)}] (t, u),$$

where  $g^{(p+1)}$  is independent of the vector of control functions  $u$  and in particular causal.

### 2.6.2 Addition

The sum of two Fliess-expansions is obviously obtained by adding the coefficients componentwise. Consider the Fliess-expansions  $y$  and  $z$  of order  $p$ .

$$\begin{aligned} y(t) &= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (y)_b \mathcal{I}_b(t) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[y_b](t) \\ z(t) &= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (z)_b \mathcal{I}_b(t) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[z_b](t) \end{aligned}$$

We prove the existence and compute the coefficients of

$$(y+z)(t) = y(t) + z(t) = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (y+z)_b \mathcal{I}_b(t) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[[y+z]_b](t)$$

by applying lemma 2.10 to every iterated integral in particular in the sum of the remainder terms. Then we get

$$(y+z)_b = (y)_b + (z)_b, \text{ for all } b \in \mathcal{M}_m, |b| \leq p.$$

### 2.6.3 Multiplication

For the computation of the product of Fliess-expansions we first analyse how iterated integrals can be multiplied. Therefore we formulate a lemma, which shows for the product of iterated integrals a behaviour similar to the

definition 1.16 of the shuffle product of their multi-indices. Depending on the type of the integral kernels and on the iteration order of the involved integrals, we then develop three theorems, which tell us how to multiply iterated integrals.

**Lemma 2.29** *For the multiplication of iterated integrals for arbitrary multi-indices  $b, c \in \mathcal{M}_m$ , indices  $\beta, \gamma \in \{0, \dots, m\}$  and functions  $G, H : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  we have*

$$\begin{aligned} \mathcal{I}_{(b, \beta)}[G](t, u) \cdot \mathcal{I}_{(c, \gamma)}[H](t, u) &= \\ &= \mathcal{I}_\beta[\mathcal{I}_b[G] \cdot \mathcal{I}_{(c, \gamma)}[H]](t, u) + \mathcal{I}_\gamma[\mathcal{I}_{(b, \beta)}[G] \cdot \mathcal{I}_c[H]](t, u). \end{aligned}$$

**Proof:** We apply lemma 2.7 to both factors and nest the right into the left integral

$$\begin{aligned} \mathcal{I}_{(b, \beta)}[G](t, u) \cdot \mathcal{I}_{(c, \gamma)}[H](t, u) &= \\ &= \int_\tau^t \mathcal{I}_b[G](s_1, u) u^\beta(s_1) ds_1 \cdot \int_\tau^t \mathcal{I}_c[H](s_2, u) u^\gamma(s_2) ds_2 \\ &= \int_\tau^t \int_\tau^t \mathcal{I}_b[G](s_1, u) \cdot \mathcal{I}_c[H](s_2, u) u^\gamma(s_2) u^\beta(s_1) ds_2 ds_1 \\ &= \int_\tau^t \int_\tau^{s_1} \mathcal{I}_b[G](s_1, u) \cdot \mathcal{I}_c[H](s_2, u) u^\gamma(s_2) u^\beta(s_1) ds_2 ds_1 \\ &\quad + \int_\tau^t \int_{s_1}^t \mathcal{I}_b[G](s_1, u) \cdot \mathcal{I}_c[H](s_2, u) u^\gamma(s_2) u^\beta(s_1) ds_2 ds_1. \end{aligned}$$

Now we interchange  $s_1$  and  $s_2$  in the second integral:

$$\begin{aligned} &= \int_\tau^t \int_\tau^{s_1} \mathcal{I}_b[G](s_1, u) \cdot \mathcal{I}_c[H](s_2, u) u^\gamma(s_2) u^\beta(s_1) ds_2 ds_1 \\ &\quad + \int_\tau^t \int_\tau^{s_1} \mathcal{I}_b[G](s_2, u) \cdot \mathcal{I}_c[H](s_1, u) u^\gamma(s_1) u^\beta(s_2) ds_2 ds_1 \\ &= \int_\tau^t \mathcal{I}_b[G](s_1, u) \int_\tau^{s_1} \mathcal{I}_c[H](s_2, u) u^\gamma(s_2) ds_2 u^\beta(s_1) ds_1 \\ &\quad + \int_\tau^t \mathcal{I}_c[H](s_1, u) \int_\tau^{s_1} \mathcal{I}_b[G](s_2, u) u^\beta(s_2) ds_2 u^\gamma(s_1) ds_1. \end{aligned}$$

Then we repeatedly apply lemma 2.7 again

$$\begin{aligned} &= \int_\tau^t \mathcal{I}_b[G](s_1, u) \cdot \mathcal{I}_{(c, \gamma)}[H](s_1, u) u^\beta(s_1) ds_1 \\ &\quad + \int_\tau^t \mathcal{I}_c[H](s_1, u) \cdot \mathcal{I}_{(b, \beta)}[G](s_1, u) u^\gamma(s_1) ds_1 \end{aligned}$$

$$= \mathcal{I}_\beta [\mathcal{I}_b [G] \cdot \mathcal{I}_{(c, \gamma)} [H]] (t, u) + \mathcal{I}_\gamma [\mathcal{I}_{(b, \beta)} [G] \cdot \mathcal{I}_c [H]] (t, u).$$

■

For iterated integrals with constant integral kernels we get exactly the [shuffle product](#) of their multi-indices. That leads us directly to the next theorem.

**Theorem 2.30 (multiplication of iterated integrals)** *For arbitrary multi-indices  $b, c \in \mathcal{M}_m$  the iterated integrals are multiplied like the shuffle-product:*

$$\mathcal{I}_b(t, u) \cdot \mathcal{I}_c(t, u) = \mathcal{I}_{b \sqcup c}(t, u).$$

**Proof:** We prove by induction over the common length  $L := |b| + |c|$  of the multi-indices.

**Initial step  $L = 0$ :**

$$\mathcal{I}_\ominus(t, u) \cdot \mathcal{I}_\ominus(t, u) = 1 = \mathcal{I}_{\ominus \sqcup \ominus}(t, u).$$

**Induction step  $L - 1 \rightarrow L$ , for all  $L > 0$ :** First we analyse the trivial case  $b = \ominus$  and  $|c| = L$ :

$$\mathcal{I}_\ominus(t, u) \cdot \mathcal{I}_c(t, u) = \mathcal{I}_c(t, u) = \mathcal{I}_{c \sqcup \ominus}(t, u).$$

The same applies for the opposite case  $|b| = L$  and  $c = \ominus$ . In the remaining case there are multi-indices  $b', c' \in \mathcal{M}_m$  and indices  $\beta, \gamma \in \{0, \dots, m\}$  with  $b = (b', \beta)$  and  $c = (c', \gamma)$ . From lemma [2.29](#) follows with  $G = H \equiv 1$ :

$$\begin{aligned} & \mathcal{I}_{(b', \beta)}(t, u) \cdot \mathcal{I}_{(c', \gamma)}(t, u) = \\ &= \int_\tau^t \mathcal{I}_{b'}(s_1, u) \cdot \mathcal{I}_{(c', \gamma)}(s_1, u) u^\beta(s_1) ds_1 \\ &+ \int_\tau^t \mathcal{I}_{(b', \beta)}(s_2, u) \cdot \mathcal{I}_{c'}(s_2, u) u^\gamma(s_2) ds_2. \end{aligned}$$

With  $|b'| + |c'| = |b| + |c| - 1 = L - 1$  we apply the induction hypothesis and get

$$= \int_\tau^t \mathcal{I}_{b' \sqcup (c', \gamma)}(s_1, u) u^\beta(s_1) ds_1 + \int_\tau^t \mathcal{I}_{(b', \beta) \sqcup c'}(s_2, u) u^\gamma(s_2) ds_2$$

$$\begin{aligned}
&= \mathcal{I}_{(b' \sqcup (c', \gamma), \beta)}(t, u) + \mathcal{I}_{((b', \beta) \sqcup c', \gamma)}(t, u) \\
&= \mathcal{I}_{(b', \beta) \sqcup (c', \gamma)}(t, u).
\end{aligned}$$

■

For non-constant integral kernels of the iterated integrals we can formulate at least the following qualitative theorem:

**Theorem 2.31 (multiplication of iterated integrals)** *For all  $b, c \in \mathcal{M}_m$ , all causal functions  $G, H : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  and all  $d \in \mathcal{M}_m^{\min\{|b|, |c|\}}$  there exist causal functions  $z_d : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$ , such that the product of two iterated integrals can be represented as a sum of iterated integrals:*

$$\mathcal{I}_b[G](t, u) \cdot \mathcal{I}_c[H](t, u) = \sum_{d \in \mathcal{M}_m^{\min\{|b|, |c|\}}} \mathcal{I}_d[z_d](t, u).$$

**Proof:** We prove by induction over the common length of the multi-indices  $L := |b| + |c|$ .

**Initial step  $L = 0$ , thus  $b = \ominus = c$ :** With  $z_\ominus := G \cdot H$  there is

$$\mathcal{I}_\ominus[G](t, u) \cdot \mathcal{I}_\ominus[H](t, u) = G(t, u) \cdot H(t, u) = \mathcal{I}_\ominus[z_\ominus](t, u).$$

**Induction step  $L - 1 \rightarrow L$ , or all  $L > 0$ :** First we analyse the case  $b = \ominus$  and  $|c| = L$ :

$$\mathcal{I}_\ominus[G](t, u) \cdot \mathcal{I}_c[H](t, u) = G(t, u) \cdot \mathcal{I}_c[H](t, u) = \mathcal{I}_\ominus[z_\ominus](t, u),$$

where  $z_\ominus := G \cdot \mathcal{I}_c[H]$ . The same applies for the symmetric case  $c = \ominus$  and  $|b| = L$ . Otherwise there exist multi-indices  $b', c' \in \mathcal{M}_m$  and indices  $\beta, \gamma \in \{0, \dots, m\}$  with  $b = (b', \beta)$  and  $c = (c', \gamma)$ . From lemma 2.29 follows:

$$\begin{aligned}
&\mathcal{I}_{(b', \beta)}[G](t, u) \cdot \mathcal{I}_{(c', \gamma)}[H](t, u) = \\
&= \mathcal{I}_\beta[\mathcal{I}_{b'}[G] \cdot \mathcal{I}_{(c', \gamma)}[H]](t, u) + \mathcal{I}_\gamma[\mathcal{I}_{(b', \beta)}[G] \cdot \mathcal{I}_{c'}[H]](t, u).
\end{aligned}$$

By the induction hypothesis there exist causal functions  $\hat{G}_d$ , for all  $d \in \mathcal{M}_m^{\min\{|b|-1, |c|\}}$  and  $\hat{H}_e$ , for all  $e \in \mathcal{M}_m^{\min\{|b|, |c|-1\}}$  with

$$\begin{aligned} &= \sum_{d \in \mathcal{M}_m^{\min\{|b|-1, |c|\}}} \mathcal{I}_\beta \left[ \mathcal{I}_d \left[ \hat{G}_d \right] \right] (t, u) + \sum_{e \in \mathcal{M}_m^{\min\{|b|, |c|-1\}}} \mathcal{I}_\gamma \left[ \mathcal{I}_e \left[ \hat{H}_e \right] \right] (t, u) \\ &= \sum_{d \in \mathcal{M}_m^{\min\{|b|-1, |c|\}}} \mathcal{I}_{(d, \beta)} \left[ \hat{G}_d \right] (t, u) + \sum_{e \in \mathcal{M}_m^{\min\{|b|, |c|-1\}}} \mathcal{I}_{(e, \gamma)} \left[ \hat{H}_e \right] (t, u). \end{aligned}$$

Now we distinguish three cases by the relative length of the multi-indices  $b$  and  $c$ . First we analyse the case  $|b| = |c|$ . Then the lengths of the multi-indices in both sums are all equal, namely  $|e| = |d| = \min\{|b| - 1, |c|\} = |b| - 1$ . Now we define for all  $f' \in \mathcal{M}_m^{|b|-1}$  and all  $\vartheta \in \{0, \dots, m\}$  the causal function

$$z_{(f', \vartheta)} := \begin{cases} \hat{G}_{f'} + \hat{H}_{f'}, & \text{for } \beta = \vartheta = \gamma, \\ \hat{G}_{f'}, & \text{for } \beta = \vartheta \neq \gamma, \\ \hat{H}_{f'}, & \text{for } \beta \neq \vartheta = \gamma, \\ 0, & \text{for } \beta \neq \vartheta \neq \gamma. \end{cases}$$

Then we get immediately the assertion:

$$\mathcal{I}_b[G](t, u) \cdot \mathcal{I}_c[H](t, u) = \sum_{f \in \mathcal{M}_m^{|b|}} \mathcal{I}_f[z_f](t, u).$$

Next we deal with the case  $|b| < |c|$ . There is  $\min\{|b| - 1, |c|\} = |b| - 1$  and  $\min\{|b|, |c| - 1\} = |b| \geq 1$ . Thus the iterated integrals of the first summand have the shorter multi-index length, namely  $|b|$ . Then we get for the second summand by using lemma 2.11

$$\begin{aligned} \sum_{e \in \mathcal{M}_m^{|b|}} \mathcal{I}_{(e, \gamma)} \left[ \hat{H}_e \right] (t, u) &= \sum_{e' \in \mathcal{M}_m^{|b|-1}} \sum_{\delta=0}^m \mathcal{I}_{(\delta, e', \gamma)} \left[ \hat{H}_{(\delta, e')} \right] (t, u) \\ &= \sum_{e' \in \mathcal{M}_m^{|b|-1}} \mathcal{I}_{(e', \gamma)} \left[ \sum_{\delta=0}^m \mathcal{I}_\delta \left[ \hat{H}_{(\delta, e')} \right] \right] (t, u). \end{aligned}$$

So we created a sum of iterated integrals with multi-index length  $|b|$  similar to the first summand. Then we define again for all multi-indices  $f' \in \mathcal{M}_m^{|b|-1}$  and all indices  $\vartheta \in \{0, \dots, m\}$  the causal function

$$z_{(f', \vartheta)} := \begin{cases} \hat{G}_{f'} + \sum_{\delta=0}^m \mathcal{I}_\delta \left[ \hat{H}_{(\delta, f')} \right], & \text{for } \beta = \vartheta = \gamma, \\ \hat{G}_{f'}, & \text{for } \beta = \vartheta \neq \gamma, \\ \sum_{\delta=0}^m \mathcal{I}_\delta \left[ \hat{H}_{(\delta, f')} \right], & \text{for } \beta \neq \vartheta = \gamma, \\ 0, & \text{for } \beta \neq \vartheta \neq \gamma, \end{cases}$$

and add up both summands

$$\mathcal{I}_b[G](t, u) \cdot \mathcal{I}_c[H](t, u) = \sum_{f \in \mathcal{M}_m^{|b|}} \mathcal{I}_f[z_f](t, u).$$

The third case with  $|b| > |c|$  runs similar. ■

Next we will formulate another theorem for the multiplication of iterated integrals. We assume one integral kernel to be an arbitrary causal function and the other one to be constant. Theorem 2.31 gives as a special case already an answer of “low order”. The next theorem formulates a “higher order” conclusion.

**Theorem 2.32 (multiplication of iterated integrals)** *For all multi-indices  $b, c \in \mathcal{M}_m$ , all causal functions  $G : [\tau, T] \times \mathcal{U}^m \rightarrow \mathbb{R}^n$  and all multi-indices  $d \in \mathcal{M}_m^{|c|}$  exist causal functions  $z_d : [\tau, T] \times \mathcal{U}^m \rightarrow \mathbb{R}^n$ , such that the product of two iterated integrals with one constant integral kernel can be transformed into a sum:*

$$\mathcal{I}_b(t, u) \cdot \mathcal{I}_c[G](t, u) = \sum_{d \in \mathcal{M}_m^{|c|}} \mathcal{I}_d[z_d](t, u).$$

**Proof:** We prove by induction over the common length of the multi-indices  $L := |b| + |c|$ .

**Initial step**  $L = 0$ , thus  $b = \ominus = c$ : With  $z_\ominus := G$  we get

$$\mathcal{I}_\ominus(t, u) \cdot \mathcal{I}_\ominus[G](t, u) = G(t, u) = \mathcal{I}_\ominus[z_\ominus](t, u).$$

**Induction step**  $L - 1 \rightarrow L$ , for all  $L > 0$ : First we analyse the case  $b = \ominus$  and  $|c| = L$ . We get

$$\mathcal{I}_\ominus(t, u) \cdot \mathcal{I}_c[G](t, u) = \mathcal{I}_c[G](t, u) = \mathcal{I}_c[z_c](t, u),$$

where  $z_c := G$ . For  $c = \ominus$  and  $|b| = L$  we get

$$\mathcal{I}_b(t, u) \cdot \mathcal{I}_\ominus[G](t, u) = \mathcal{I}_b(t, u) \cdot G(t, u) = \mathcal{I}_\ominus[z_\ominus](t, u),$$

with  $z_\ominus := \mathcal{I}_b(t, u) \cdot G$ .

Otherwise there are multi-indices  $b', c' \in \mathcal{M}$  and indices  $\beta, \gamma \in \{0, \dots, m\}$  with  $b = (b', \beta)$  and  $c = (c', \gamma)$ . We get from lemma 2.29

$$\begin{aligned} \mathcal{I}_{(b', \beta)}(t, u) \cdot \mathcal{I}_{(c', \gamma)}[G](t, u) &= \\ &= \mathcal{I}_\beta [\mathcal{I}_{b'} \cdot \mathcal{I}_{(c', \gamma)}[G]](t, u) + \mathcal{I}_\gamma [\mathcal{I}_{(b', \beta)} \cdot \mathcal{I}_{c'}[G]](t, u). \end{aligned}$$

Applying the induction hypothesis there exist causal functions  $\hat{G}_d, d \in \mathcal{M}_m^{|c|}$  and  $\hat{H}_e, e \in \mathcal{M}_m^{|c|-1}$  for replacing products by sums

$$\begin{aligned} &= \sum_{d \in \mathcal{M}_m^{|c|}} \mathcal{I}_\beta [\mathcal{I}_d [\hat{G}_d]](t, u) + \sum_{e \in \mathcal{M}_m^{|c|-1}} \mathcal{I}_\gamma [\mathcal{I}_e [\hat{H}_e]](t, u) \\ &= \sum_{d' \in \mathcal{M}_m^{|c|-1}} \sum_{\delta=0}^m \mathcal{I}_\beta [\mathcal{I}_{(\delta, d')} [\hat{G}_{(\delta, d')}]](t, u) + \sum_{e \in \mathcal{M}_m^{|c|-1}} \mathcal{I}_\gamma [\mathcal{I}_e [\hat{H}_e]](t, u) \\ &= \sum_{d' \in \mathcal{M}_m^{|c|-1}} \mathcal{I}_{(d', \beta)} \left[ \sum_{\delta=0}^m \mathcal{I}_\delta [\hat{G}_{(\delta, d')}] \right](t, u) + \sum_{e \in \mathcal{M}_m^{|c|-1}} \mathcal{I}_{(e, \gamma)} [\hat{H}_e](t, u). \end{aligned}$$

Next we define for all multi-indices  $f' \in \mathcal{M}_m^{|c|-1}$  and all integer numbers  $\vartheta \in \{0, \dots, m\}$  the causal function

$$z_{(f', \vartheta)} := \begin{cases} \sum_{\delta=0}^m \mathcal{I}_\delta [\hat{G}_{(\delta, f')}] + \hat{H}_{f'}, & \text{for } \beta = \vartheta = \gamma, \\ \sum_{\delta=0}^m \mathcal{I}_\delta [\hat{G}_{(\delta, f')}] , & \text{for } \beta = \vartheta \neq \gamma, \\ \hat{H}_{f'}, & \text{for } \beta \neq \vartheta = \gamma, \\ 0, & \text{for } \beta \neq \vartheta \neq \gamma. \end{cases}$$

Then we get the assertion:

$$\mathcal{I}_b(t, u) \cdot \mathcal{I}_c[G](t, u) = \sum_{f \in \mathcal{M}_m^{|c|}} \mathcal{I}_f[z_f](t, u).$$

■

Now we know how to multiply iterated integrals with causal and with constant integral kernels and also how the mixed multiplications can be represented. For the multiplication with non-constant kernels we are only interested in the existence of the representation of the sum of “high” order.

Next we will use the last three theorems to multiply complete Fliess-expansions including their remainder terms. We consider two functions  $y, z \in [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$ . Their Fliess-expansions of order  $p$  are assumed to exist, that is there exist coefficients  $(y)_b, (z)_b \in \mathbb{R}^n$ , for all  $b \in \mathcal{M}_m$  with

$|b| \leq p$ , and there exist causal functions  $y_b, z_b \in [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$ , for all  $b \in \mathcal{M}_m^{p+1}$ , such that

$$\begin{aligned} y(t, u) &= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (y)_b \mathcal{I}_b(t, u) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[y_b](t, u), \\ z(t, u) &= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (z)_b \mathcal{I}_b(t, u) + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[z_b](t, u). \end{aligned}$$

**Theorem 2.33 (multiplication of Fliess expansions)** *There exists a Fliess-expansion for the product function  $yz : [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$ . Its coefficients are*

$$(yz)_b = \sum_{k=0}^{|b|} \sum_{i=1}^{C(|b|, k)} (y)_{\mathcal{K}(b, k, i)} (z)_{\widehat{\mathcal{K}}(b, k, i)}, \quad (2.12)$$

for all  $b \in \mathcal{M}_m$ .

**Proof:** We multiply the two Fliess-expansions for  $y$  and  $z$ . For simpler notation we define

$$\begin{aligned} R_1(t, u) &:= \sum_{i=p+1}^{2p} \sum_{k=i-p}^p \sum_{b \in \mathcal{M}_m^{i-k}} \sum_{c \in \mathcal{M}_m^k} (y)_b (z)_c \mathcal{I}_b(t, u) \mathcal{I}_c(t, u), \\ R_2(t, u) &:= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{c \in \mathcal{M}_m^{p+1}} (y)_b \mathcal{I}_b(t, u) \mathcal{I}_c[z_c](t, u), \\ R_3(t, u) &:= \sum_{i=0}^p \sum_{c \in \mathcal{M}_m^i} \sum_{b \in \mathcal{M}_m^{p+1}} (z)_c \mathcal{I}_c(t, u) \mathcal{I}_b[y_b](t, u), \\ R_4(t, u) &:= \sum_{b \in \mathcal{M}_m^{p+1}} \sum_{c \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[y_b](t, u) \mathcal{I}_c[z_c](t, u). \end{aligned}$$

So we can write down the product:

$$\begin{aligned} y(t, u)z(t, u) &= \sum_{i=0}^p \sum_{j=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{c \in \mathcal{M}_m^j} (y)_b (z)_c \mathcal{I}_b(t, u) \mathcal{I}_c(t, u) + \\ &\quad + R_2(t, u) + R_3(t, u) + R_4(t, u). \end{aligned}$$



We apply a Cauchy-like product to the two outer sums and replace the product of the iterated integrals by the shuffle-product of their multi-indices (cp. theorem 2.30)

$$\begin{aligned}
&= \sum_{i=0}^p \sum_{k=0}^i \sum_{b \in \mathcal{M}_m^{i-k}} \sum_{c \in \mathcal{M}_m^k} (y)_b (z)_c \mathcal{I}_{b \sqcup c}(t, u) + R_1(t, u) + \\
&\quad + R_2(t, u) + R_3(t, u) + R_4(t, u).
\end{aligned}$$

First we analyse only the first sum:

$$\begin{aligned}
&\sum_{i=0}^p \sum_{k=0}^i \sum_{b \in \mathcal{M}_m^k} \sum_{c \in \mathcal{M}_m^{i-k}} (y)_b (z)_c \mathcal{I}_{b \sqcup c}(t, u) = \\
&= \sum_{i=0}^p \sum_{d \in \mathcal{M}_m^i} \sum_{k=0}^i (y)_{L_k(d)} (z)_{R_{i-k}(d)} \mathcal{I}_{L_k(d) \sqcup R_{i-k}(d)}(t, u),
\end{aligned}$$

where the decomposition operators  $L$  and  $R$  from definition 1.13 are used. Then we insert the explicit version for the shuffle-product (cp. theorem 1.25):

$$= \sum_{i=0}^p \sum_{d \in \mathcal{M}_m^i} \sum_{k=0}^i \sum_{j=1}^{C(i,k)} (y)_{L_k(d)} (z)_{R_{i-k}(d)} \mathcal{I}_{\mathcal{A}(L_k(d), R_{i-k}(d), j)}(t, u).$$

We resort the sums using lemma 1.22 and the properties

$$\begin{aligned}
L_k \left( \left( \mathcal{K}(d, k, j), \widehat{\mathcal{K}}(d, k, j) \right) \right) &= \mathcal{K}(d, k, j) \\
R_{|d|-k} \left( \left( \mathcal{K}(d, k, j), \widehat{\mathcal{K}}(d, k, j) \right) \right) &= \widehat{\mathcal{K}}(d, k, j)
\end{aligned}$$

and get rid of the  $L$  and  $R$  operators:

$$= \sum_{i=0}^p \sum_{k=0}^i \sum_{j=1}^{C(i,k)} \sum_{d \in \mathcal{M}_m^i} (y)_{\mathcal{K}(d, k, j)} (z)_{\widehat{\mathcal{K}}(d, k, j)} \mathcal{I}_{\mathcal{A}(\mathcal{K}(d, k, j), \widehat{\mathcal{K}}(d, k, j), j)}(t, u).$$

Then it follows from theorem 1.26 and another reordering of the sums

$$\begin{aligned}
&= \sum_{i=0}^p \sum_{d \in \mathcal{M}_m^i} \sum_{k=0}^i \sum_{j=1}^{C(i,k)} (y)_{\mathcal{K}(d, k, j)} (z)_{\widehat{\mathcal{K}}(d, k, j)} \mathcal{I}_d(t, u) \\
&= \sum_{i=0}^p \sum_{d \in \mathcal{M}_m^i} (yz)_d \mathcal{I}_d(t, u),
\end{aligned}$$

$$\text{with } (yz)_d := \sum_{k=0}^{|d|} \sum_{j=1}^{C(|d|,k)} (y)_{\mathcal{K}(d,k,j)} (z)_{\widehat{\mathcal{K}}(d,k,j)}.$$

This is already the recursion formula for the coefficients  $(yz)_d$ ,  $|d| \leq p$ . So it remains to prove that  $R_1 + R_2 + R_3 + R_4$  represents a remainder of a FlieSS-expansion. We have:

$$R_1(t, u) = \sum_{i=p+1}^{2p} \sum_{k=i-p}^p \sum_{b \in \mathcal{M}_m^{i-k}} \sum_{c \in \mathcal{M}_m^k} (y)_b (z)_c \mathcal{I}_b(t, u) \mathcal{I}_c(t, u) =$$

As we did above we get

$$\begin{aligned} &= \sum_{i=p+1}^{2p} \sum_{d \in \mathcal{M}_m^i} \sum_{k=i-p}^p \sum_{j=1}^{C(i,k)} (y)_{\mathcal{K}(d,k,j)} (z)_{\widehat{\mathcal{K}}(d,k,j)} \mathcal{I}_d(t, u) \\ &= \sum_{i=p+1}^{2p} \sum_{e \in \mathcal{M}_m^{p+1}} \sum_{f \in \mathcal{M}_m^{i-(p+1)}} \sum_{k=i-p}^p \sum_{j=1}^{C(i,k)} (y)_{\mathcal{K}((f, e), k, j)} (z)_{\widehat{\mathcal{K}}((f, e), k, j)} \mathcal{I}_{(f, e)}(t, u). \end{aligned}$$

From lemma 2.11 follows

$$\begin{aligned} &= \sum_{e \in \mathcal{M}_m^{p+1}} \sum_{i=p+1}^{2p} \sum_{f \in \mathcal{M}_m^{i-(p+1)}} \sum_{k=i-p}^p \sum_{j=1}^{C(i,k)} (y)_{\mathcal{K}((f, e), k, j)} (z)_{\widehat{\mathcal{K}}((f, e), k, j)} \mathcal{I}_e [\mathcal{I}_f] (t, u) \\ &= \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e [C_e^1] (t, u), \\ \text{with } C_e^1 &= \sum_{i=p+1}^{2p} \sum_{f \in \mathcal{M}_m^{i-(p+1)}} \sum_{k=i-p}^p \sum_{j=1}^{C(i,k)} (y)_{\mathcal{K}((f, e), k, j)} (z)_{\widehat{\mathcal{K}}((f, e), k, j)} \mathcal{I}_f. \end{aligned}$$

The causality of  $\mathcal{I}_f$  carries forward to  $C_e^1$ .

We continue with  $R_2$ :

$$R_2(t, u) = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{c \in \mathcal{M}_m^{p+1}} (y)_b \mathcal{I}_b(t, u) \mathcal{I}_c[z_c] (t, u)$$

From theorem 2.32 follows the existence of causal functions  $\hat{C}_e^2 \in [\tau, T] \times \mathcal{U}^m \rightarrow \mathbb{R}^n$ ,  $e \in \mathcal{M}_m^{p+1}$  with

$$= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{c \in \mathcal{M}_m^{p+1}} (y)_b \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e [\hat{C}_e^2] (t, u)$$

$$\begin{aligned}
&= \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e \left[ \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{c \in \mathcal{M}_m^{p+1}} (y)_b \hat{C}_e^2 \right] (t, u) \\
&= \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e [C_e^2] (t, u), \\
&\text{with } C_e^2 = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{c \in \mathcal{M}_m^{p+1}} (y)_b \hat{C}_e^2 \\
&= (m+1)^{(p+1)} \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (y)_b \hat{C}_e^2.
\end{aligned}$$

The functions  $C_e^2[\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  are causal. Next we analyse  $R_3$ :

$$R_3(t, u) = \sum_{i=0}^p \sum_{c \in \mathcal{M}_m^i} \sum_{b \in \mathcal{M}_m^{p+1}} (y)_c \mathcal{I}_c(t, u) \mathcal{I}_b[z_b](t, u).$$

From theorem 2.32 follows the existence of causal functions  $\hat{C}_e^3 \in [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$ ,  $e \in \mathcal{M}_m^{p+1}$  with

$$\begin{aligned}
&= \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{c \in \mathcal{M}_m^{p+1}} (y)_b \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e [\hat{C}_e^3] (t, u) \\
&= \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e \left[ \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{c \in \mathcal{M}_m^{p+1}} (y)_b \hat{C}_e^3 \right] (t, u) \\
&= \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e [C_e^3] (t, u), \\
&\text{with } C_e^3 = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{c \in \mathcal{M}_m^{p+1}} (y)_b \hat{C}_e^3 \\
&= (m+1)^{(p+1)} \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (y)_b \hat{C}_e^3.
\end{aligned}$$

The functions  $C_e^3[\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  are causal. Next we analyse  $R_4$ :

$$R_4(t, u) := \sum_{b \in \mathcal{M}_m^{p+1}} \sum_{c \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[y_b](t, u) \mathcal{I}_c[z_c](t, u).$$

From theorem 2.31 follows the existence of causal functions  $\hat{C}_e^4 \in [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  with

$$= \sum_{b \in \mathcal{M}_m^{p+1}} \sum_{c \in \mathcal{M}_m^{p+1}} \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e [\hat{C}_e^4] (t, u)$$

$$\begin{aligned}
&= \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e \left[ \sum_{b \in \mathcal{M}_m^{p+1}} \sum_{c \in \mathcal{M}_m^{p+1}} \hat{C}_e^4 \right] (t, u) \\
&= \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e [C_e^4] (t, u), \\
&\text{with } C_e^4 = (m+1)^{2(p+1)} \hat{C}_e^4.
\end{aligned}$$

The functions  $C_e^4[\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  are causal. Now we sum up the four remainder terms and get with  $C_e := \sum_{i=0}^4 C_e^i$ , for all  $e \in \mathcal{M}_m^{p+1}$ ,

$$R_1(t, u) + R_2(t, u) + R_3(t, u) + R_4(t, u) = \sum_{e \in \mathcal{M}_m^{p+1}} \mathcal{I}_e [C_e] (t, u),$$

with  $C_e \in [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$  are causal functions. So the product  $yz$  can be represented as a Fliess-expansion of order  $p$  (cp. definition 2.16). ■

#### 2.6.4 Taylor-expansions of Fliess-expansions

The ability to multiply Fliess-expansions gives us the possibility to insert a Fliess-expansion into a Taylor-expansion. For simplicity we will only show the one-dimensional case.

Consider a function  $g : \mathbb{R} \rightarrow \mathbb{R}$ , whose Taylor-expansion we assume to be known.

$$\begin{aligned}
g(y) &= \sum_{i=0}^p (g)_i (y - y_0)^i + R_g^{p+1}(y) \\
&\text{with } R_g^{p+1}(y) := \int_{y_0}^y \int_{y_0}^{y_1} \cdots \int_{y_0}^{y_p} g^{(p+1)}(y_{p+1}) \, dy_{p+1} \cdots dy_2 \, dy_1.
\end{aligned} \tag{2.13}$$

Let  $y = y(t, u)$  be the Fliess-expansion of order  $p$ , which we insert into  $g$ . Therefore we choose  $y_0 = y(\tau, u)$ . We will show that the result  $g(y)$  can be represented as Fliess-expansion of order  $p$ . We know from the multiplication (cp. theorem 2.33) that  $(y - y_0)^i$ , for  $i = 0, \dots, p$ , are Fliess-expansions of order  $p$ . We compute the coefficients of the result by using the multiplication and addition formulas. The part that is left to prove is the representation of  $R_g^{p+1}(y)$  as a Fliess-remainder. Therefore we first formulate a technical lemma.

**Lemma 2.34** *Let  $f$  be a causal function of order  $i$ , that is there exist for all  $b \in \mathcal{M}_m^i$  causal functions  $\eta_b$ , such that*

$$f(t, u) = \sum_{b \in \mathcal{M}_m^i} \mathcal{I}_b [\eta_b] (t, u),$$

and  $y$  is a Fliess-expansion of order  $p$ . Then there exist for all  $c \in \mathcal{M}_m^{i+1}$  causal functions  $\xi_c : [\tau, T] \times \mathcal{U}^m \rightarrow \mathbb{R}$  with

$$\int_{\tau}^t f(s, u) \dot{y}(t, u) \, ds = \sum_{c \in \mathcal{M}_m^{i+1}} \mathcal{I}_c [\xi_c] (t, u).$$

**Proof:** The time-derivative of the Fliess-expansion  $y$  is

$$\begin{aligned} \dot{y}(t, u) &= \sum_{j=0}^{p-1} \sum_{d \in \mathcal{M}_m^j} \sum_{\gamma=0}^m (y)_{(d, \gamma)} \mathcal{I}_d(t, u) u^{\gamma}(t) \\ &\quad + \sum_{d \in \mathcal{M}_m^p} \sum_{\gamma=0}^m \mathcal{I}_d [y_{(d, \gamma)}] (t, u) u^{\gamma}(t). \end{aligned}$$

We multiply the expansion for  $\dot{y}$  with the sum-representation of  $f$ . Thereby meet the iterated integrals  $\mathcal{I}_b [\eta_b]$  and  $\mathcal{I}_d$  in the first summand, where  $|b| = i$  and  $|d| = j \geq 0$ . We multiply them using lemma 2.32. Then there exist causal functions  $\zeta_e$  for  $e \in \mathcal{M}_m^i$ , such that

$$\mathcal{I}_b [\eta_b] (t, u) \mathcal{I}_d(t, u) = \sum_{e \in \mathcal{M}_m^i} \mathcal{I}_e [\zeta_e] (t, u).$$

The  $\zeta_e$  depend on  $b, d$  and  $\gamma$ . The second summand multiplies the integrals  $\mathcal{I}_b [\eta_b]$  and  $\mathcal{I}_d [y_{(d, \gamma)}]$ , with  $|b| = i < p$  and  $|d| = p$ . Then lemma 2.31 guarantees us the existence of causal functions  $\vartheta_e$  for  $e \in \mathcal{M}_m^i$ , such that

$$\mathcal{I}_b [\eta_b] (t, u) \mathcal{I}_d [y_{(d, \gamma)}] (t, u) = \sum_{e \in \mathcal{M}_m^i} \mathcal{I}_e [\vartheta_e] (t, u).$$

The functions  $\vartheta_e$  again depend on the indices  $b, d$  and  $\gamma$ . We execute the summation over  $j$  (only in the first summand),  $b$  and  $d$ . Then there exist causal functions  ${}_{\gamma}\zeta_e$  and  ${}_{\gamma}\vartheta_e$ , such that the integral can be represented as

$$\begin{aligned} \int_{\tau}^t f(s, u) \dot{y}(t, u) \, ds &= \sum_{\gamma=0}^m \sum_{e \in \mathcal{M}_m^i} \int_{\tau}^t \mathcal{I}_e [{}_{\gamma}\zeta_e] (s, u) u^{\gamma}(s, u) \, ds \\ &\quad + \sum_{\gamma=0}^m \sum_{e \in \mathcal{M}_m^i} \int_{\tau}^t \mathcal{I}_e [{}_{\gamma}\vartheta_e] (s, u) u^{\gamma}(s, u) \, ds \\ &= \sum_{e \in \mathcal{M}_m^i} \sum_{\gamma=0}^m \int_{\tau}^t \mathcal{I}_e [{}_{\gamma}\zeta_e + {}_{\gamma}\vartheta_e] (s, u) u^{\gamma}(s, u) \, ds \\ &= \sum_{e \in \mathcal{M}_m^i} \sum_{\gamma=0}^m \mathcal{I}_{(e, \gamma)} [{}_{\gamma}\zeta_e + {}_{\gamma}\vartheta_e] (t, u) \\ &= \sum_{e \in \mathcal{M}_m^{i+1}} \mathcal{I}_e [\zeta_e + \vartheta_e] (t, u), \end{aligned}$$

with the causal integral kernel  $\zeta_e + \vartheta_e$ . ■

The next lemma asserts, that the remainder of the Taylor-expansion results in the remainder of a Fliess-expansion with the same order.

**Lemma 2.35** *For the remainder term  $R_g^{p+1}$  exist causal functions  $\xi_b$ ,  $b \in \mathcal{M}_m^{p+1}$ , such that*

$$R_g^{p+1}(y(t, u)) = \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[\xi_b](t, u).$$

**Proof:** We deduce from the definition of the remainder term (equation 2.13):

$$\begin{aligned} R_g^{p+1}(y(t, u)) &= \\ &= \int_{y_0}^{y(t, u)} \int_{y_0}^{y_1} \cdots \int_{y_0}^{y_p} g^{(p+1)}(y_{p+1}) dy_{p+1} \cdots dy_2 dy_1 \\ &= \int_{\tau}^t \int_{y_0}^{y(s_1, u)} \cdots \int_{y_0}^{y_p} g^{(p+1)}(y_{p+1}) dy_{p+1} \cdots dy_2 \dot{y}(s_1, u) ds_1 \\ &\vdots \\ &= \int_{\tau}^t \int_{\tau}^{s_1} \cdots \int_{\tau}^{s_p} g^{(p+1)}(y(s_{p+1}, u)) \dot{y}(s_{p+1}, u) ds_{p+1} \cdots \\ &\quad \cdots \dot{y}(s_2, u) ds_2 \dot{y}(s_1, u) ds_1. \end{aligned}$$

Thus it is a  $(p+1)$ -times integral of the causal function  $g^{(p+1)}(y)$  multiplied in every integration with the derivative of the Fliess-expansion of  $y$  of order  $p$ . The assertion follows from the multiple application of lemma 2.34. ■

The  $n$ -dimensional Taylor-expansion has the same structure. It consists of powers of the components of  $(y - (y)_{\odot})$  and its remainder can be shown to be a Fliess-remainder like in the one-dimensional case. Its proof is very voluminous and gives no new ideas, hence we do not perform it here.

### 2.6.5 Division

For the component-wise division of Fliess-expansion we first verify the existence of the quotient. Then we develop a recursion formula out of the product rule (theorem. 2.33)

We consider two functions  $y, z \in [\tau, T] \times \mathcal{U}^m \longrightarrow \mathbb{R}^n$ , where  $z(t, u) \neq 0$  for all  $t \in [\tau, T]$  and all  $u \in \mathcal{U}^m$ . We assume the existence of their Fliess-expansions of order  $p$ .

$$y(t) = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (y)_b \mathcal{I}_b(t, u) + R_y^{p+1}(t, u)$$

$$z(t) = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (z)_b \mathcal{I}_b(t, u) + R_z^{p+1}(t, u).$$

For the function  $\frac{1}{z}$  we know from analysis the Taylor expansion of order  $p$  at the point  $z_0 := (z)_\ominus$ :

$$\frac{1}{z} = \sum_{i=0}^p \frac{(-1)^i}{z_0^{i+1}} (z - z_0)^i + R_{\frac{1}{z}}^{p+1}(z).$$

We insert the Fliess-expansion of  $z$  into the Taylor expansion and get, as section 2.6.4 explains, again a Fliess-expansion for  $\frac{1}{z}$ . The remainder  $R_{\frac{1}{z}}^{p+1}(z)$  turns together with the remainders from the powers into a Fliess remainder (cp. lemma 2.35). So the coefficients of the product

$$\frac{y}{z}(t) = \frac{y(t)}{z(t)} = \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \left(\frac{y}{z}\right)_b \mathcal{I}_b(t) + R_{\frac{y}{z}}^{p+1}(t, u)$$

can be computed with the product rule for the powers  $(z - z_0)^i$ ,  $i = 0, \dots, p$  and the multiplication  $y * \frac{1}{z}$ , which leads again to Fliess-expansion. After we have proved the existence of the Fliess-expansion of order  $p$  for  $\frac{y}{z}$  we now give a recursion formula for its coefficients.

**Theorem 2.36 (division of Fliess expansions)** *The coefficients of the quotient of two Fliess-expansions  $y$  and  $z$  can be computed with the recursion formula:*

$$\left(\frac{y}{z}\right)_b = \frac{1}{(z)_\ominus} \left( (y)_b - \sum_{k=0}^{|b|-1} \sum_{i=1}^{C(|b|,k)} \left(\frac{y}{z}\right)_{\mathcal{K}(b,k,i)} (z)_{\widehat{\mathcal{K}}(b,k,i)} \right)$$

**Proof:** We formulate the product rule (theorem 2.33) for  $y = \frac{y}{z} * z$  and solve it for  $\left(\frac{y}{z}\right)_b$ .

$$\begin{aligned} (y)_b &= \sum_{k=0}^{|b|} \sum_{i=1}^{C(|b|,k)} \left(\frac{y}{z}\right)_{\mathcal{K}(b,k,i)} (z)_{\widehat{\mathcal{K}}(b,k,i)} \\ &= \left(\frac{y}{z}\right)_{\mathcal{K}(b,|b|,l)} (z)_{\widehat{\mathcal{K}}(b,|b|,l)} + \sum_{k=0}^{|b|-1} \sum_{i=1}^{C(|b|,k)} \left(\frac{y}{z}\right)_{\mathcal{K}(b,k,i)} (z)_{\widehat{\mathcal{K}}(b,k,i)} \end{aligned}$$

$$= \left(\frac{y}{z}\right)_b (z)_\ominus + \sum_{k=0}^{|b|-1} \sum_{i=1}^{C(|b|,k)} \left(\frac{y}{z}\right)_{\mathcal{K}(b,k,i)} (z)_{\widehat{\mathcal{K}}(b,k,i)}$$

We solve for  $\left(\frac{y}{z}\right)_b$  and get the assertion. ■



## Chapter 3

# Numerical Algorithm

In this chapter we formulate the numerical algorithm to enclose all solutions of the initial value problem and the initial interval problem on a given time interval. The arithmetic operations we defined for Fliess-expansions in the last chapter enable us to construct another Fliess-expansion for the solution out of the systems right hand side. The uniqueness allows us to compare their coefficients and we get conditional equations for them. This gives a concept of automatic differentiating for Lie-derivatives.

Next we will develop the algorithm to find an enclosure for the remainder term. This is the first time we take into account the control functions. As a natural condition we need to restrict their ranges. For the remainder computation we need an a priori enclosure for all solution in the time interval. We will give a criterion to identify an interval as an enclosure for all solutions.

Then we interpret the initial interval as an uncertainty in the initial condition and linearise the algorithm for computing the coefficients of the solution to get the error propagation. The remainder's enclosure can be easily extended to all initial values. At the end of this section we formulate the algorithm and give a one dimensional example to illustrate all intervals, which add up to the enclosure of all solutions for all control functions in the given range.

### 3.1 Coefficients for the Solution

We know now that the solution  $\lambda(\cdot, \tau, x_0, u)$  can be represented as a Fliess-expansion of arbitrary order. As well the Fliess-expansion of the right hand side vector fields  $f^\alpha(\cdot, \lambda(\cdot, \tau, x_0, u))$  can be computed out of  $\lambda$ . With the integral equation we can construct another Fliess-expansion for the solution.

With the uniqueness of Fliess-expansions we can compare the coefficients and get recursion formulas for the solution.

**Theorem 3.1 (differential conditions for Fliess-expansions)**

*The coefficients of the Fliess-expansions of the solution  $\lambda$  and the right hand sides  $f^\alpha(\cdot, \lambda)$ ,  $\alpha \in \{1, \dots, m\}$  fulfil the following recursion formulas:*

$$\begin{aligned} (\lambda)_\ominus &= x(\tau), \\ (\lambda)_{(b, \beta)} &= \begin{cases} 0, & \text{for } \beta = 0 \\ (f^\beta)_b, & \text{for } \beta \in \{1, \dots, m\}, \end{cases} \end{aligned}$$

for all  $b \in \mathcal{M}_m$ , with  $|b| \leq p-1$  and  $\beta \in \{0, \dots, m\}$ .

**Proof:** We prove theorem 3.1 first for a Fliess-expansion for the solution of order 0. It coincides with the integral representation of the control system 1.1.

$$\lambda = (\lambda)_\ominus + \sum_{\alpha=0}^m \mathcal{I}_\alpha [\lambda_\alpha], \quad (3.1)$$

where from theorem 2.12 and equation (2.8) follows for  $p = 0$ :

$$\begin{aligned} (\lambda)_\ominus &= \lambda(\tau, u) = x_0 \\ \lambda(t, u)_\alpha &= \begin{cases} 0, & \text{for } \alpha = 0, \\ f^\alpha(t, \lambda(t, u)), & \text{for } \alpha \in \{1, \dots, m\}. \end{cases} \end{aligned}$$

Next we insert this coefficients and the Fliess-expansion of order  $p$  for the vector fields  $f^\alpha$ ,  $\alpha \in \{1, \dots, m\}$ , into equation (3.1). They exist by theorem 2.13. We define  $f^0 \equiv 0$  and all coefficients  $(f^0)_b = 0$ , for  $b \in \mathcal{M}_m$ ,  $|b| \leq p$ , and  $f_b^0 \equiv 0$  for  $b \in \mathcal{M}_m^{p+1}$ . For simpler notation we skip the argument  $(t, u)$  of  $\lambda$  and of the iterated integrals. We remember the relation  $\mathcal{I}_b[\mathcal{I}_c] = \mathcal{I}_{(c, b)}$  for all multi-indices  $b, c \in \mathcal{M}_m$  (cp. lemma 2.11).

$$\begin{aligned} \lambda &= x_0 + \sum_{\alpha=0}^m \mathcal{I}_\alpha [f^\alpha] \\ &= x_0 + \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \sum_{\alpha=0}^m (f^\alpha)_b \mathcal{I}_{(b, \alpha)} + \sum_{b \in \mathcal{M}_m^{p+1}} \sum_{\alpha=0}^m \mathcal{I}_{(b, \alpha)} [f_b^\alpha] \end{aligned}$$

$$\begin{aligned}
&= x_0 + \sum_{i=1}^p \sum_{b \in \mathcal{M}_m^{i-1}} \sum_{\alpha=0}^m (f^\alpha)_b \mathcal{I}_{(b, \alpha)} + \sum_{b \in \mathcal{M}_m^p} \sum_{\alpha=0}^m (f^\alpha)_b \mathcal{I}_{(b, \alpha)} \\
&\quad + \sum_{\beta=0}^m \sum_{b' \in \mathcal{M}_m^p} \sum_{\alpha=0}^m \mathcal{I}_{(\beta, b', \alpha)} \left[ f_{(\beta, b')}^\alpha \right] \\
&= x_0 + \sum_{i=1}^p \sum_{b \in \mathcal{M}_m^i} \left( f^{R_1(b)} \right)_{L_{i-1}(b)} \mathcal{I}_b + \sum_{b \in \mathcal{M}_m^{p+1}} \left( f^{R_1(b)} \right)_{L_{i-1}(b)} \mathcal{I}_b \\
&\quad + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b \left[ \sum_{\beta=0}^m \mathcal{I}_\beta \left[ f_{(\beta, L_{i-1}(b))}^{R_1(b)} \right] \right] \\
&= x_0 + \sum_{i=1}^p \sum_{b \in \mathcal{M}_m^i} \left( f^{R_1(b)} \right)_{L_{i-1}(b)} \mathcal{I}_b \\
&\quad + \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b \left[ \left( f^{R_1(b)} \right)_{L_{i-1}(b)} + \sum_{\beta=0}^m \mathcal{I}_\beta \left[ f_{(\beta, L_{i-1}(b))}^{R_1(b)} \right] \right]
\end{aligned}$$

Consequently we constructed a Fliess expansion of order  $p$  for the solution  $\lambda$ . By uniqueness of the coefficients (theorem 2.28) follows for all  $b \in \mathcal{M}_m$  with  $1 \leq |b| \leq p$

$$(\lambda)_b = \begin{cases} 0, & \text{for } R_1(b) = (0), \\ (f^{R_1(b)})_{L_{i-1}(b)}, & \text{otherwise,} \end{cases}$$

and thereby the assertion. ■

With this theorem we can now formulate the recursive algorithm to compute the coefficients of the solution's Fliess-expansion. We formulate the algorithm for both the initial value and the initial interval problem.

**Algorithm 3.2 (coefficients of the solution)**

**Step 0:** We start with an initial value  $x_0$  or an initial interval  $[x_0]$  as the coefficient with empty index  $(\lambda)_\ominus$ .

**Step  $i$ , for  $i = 1, \dots, p$ :** We compute recursively for all  $b \in \mathcal{M}_m^i$  the coefficients  $(\lambda(\tau, x_0))_b$  or  $(\lambda(\tau, [x_0]))_b$  of the solution's Fliess-expansion with theorem 3.1.

In the next example we make ourselves familiar with the recursion algorithm 3.2. We use the same control system as in section 2.14.

**Example 3.3 (coefficients of the solution)** We perform algorithm 3.2 for the system equation of example 2.14:

$$\dot{x} = u - x^2, \quad x(0) = 0. \quad (3.2)$$

The data in names of definition 1.1 are:  $m = 2$ ,  $f^1 = -x^2$  and  $f^2 = 1$ .

For the computation of the coefficients  $(f^\alpha)_b$  out of the coefficients  $(\lambda)_c$ ,  $|c| \leq |b|$ , of the solution  $\lambda := \lambda(\cdot, 0, 0, u)$  we write down the coefficient formulas for the basic operations, which compose the right hand sides. The function  $f^1$  is essentially the multiplication formula (2.12):

$$\begin{aligned} (f^1(\lambda))_\ominus &= -(\lambda)_\ominus^2, \\ (f^1(\lambda))_{(\alpha)} &= -2(\lambda)_{(\alpha)}(\lambda)_\ominus, \\ (f^1(\lambda))_{(\alpha, \beta)} &= -2(\lambda)_{(\alpha, \beta)}(\lambda)_\ominus - 2(\lambda)_{(\alpha)}(\lambda)_{(\beta)}, \\ (f^1(\lambda))_{(\alpha, \beta, \gamma)} &= -2(\lambda)_{(\alpha, \beta, \gamma)}(\lambda)_\ominus - 2(\lambda)_{(\alpha, \gamma)}(\lambda)_{(\beta)} \\ &\quad - 2(\lambda)_{(\alpha, \beta)}(\lambda)_{(\gamma)} - 2(\lambda)_{(\alpha)}(\lambda)_{(\beta, \gamma)}, \\ (f^1(\lambda))_{(\alpha, \beta, \gamma, \delta)} &= -2(\lambda)_{(\alpha, \beta, \gamma, \delta)}(\lambda)_\ominus - 2(\lambda)_{(\alpha, \gamma, \delta)}(\lambda)_{(\beta)} \\ &\quad - 2(\lambda)_{(\alpha, \beta, \delta)}(\lambda)_{(\gamma)} - 2(\lambda)_{(\alpha, \delta)}(\lambda)_{(\beta, \gamma)} \\ &\quad - 2(\lambda)_{(\alpha, \beta, \gamma)}(\lambda)_{(\delta)} - 2(\lambda)_{(\alpha, \gamma)}(\lambda)_{(\beta, \delta)} \\ &\quad - 2(\lambda)_{(\alpha, \beta)}(\lambda)_{(\gamma, \delta)} - 2(\lambda)_{(\alpha)}(\lambda)_{(\beta, \gamma, \delta)}. \end{aligned}$$

The function  $f^2$  is a constant function (cp. section 2.6.1):

$$(f^2(\lambda))_b = \begin{cases} 1, & b = \ominus, \\ 0, & \text{otherwise.} \end{cases}$$

Now we can compute the coefficients for  $\lambda$  with algorithm 3.2. We start with  $(\lambda)_\ominus = x_0 = 0$  and compute then recursively  $(\lambda)_b$ , for  $|b| = 1, \dots, 5$ . We list only the coefficients, which are non-zero:

$$\begin{aligned} (\lambda)_{(2)} &= (f^2(\lambda))_\ominus = 1, \\ (\lambda)_{(2, 2, 1)} &= (f^1(\lambda))_{(2, 2)} \end{aligned}$$

$$\begin{aligned}
&= -2((\lambda)_{(2,2)})(\lambda)_{\ominus} - 2(\lambda)_{(2)}(\lambda)_{(2)} = -2, \\
(\lambda)_{(2,2,2,1,1)} &= (f^1(\lambda))_{(2,2,2,1)} \\
&= -2(\lambda)_{(2,2,2,1)}(\lambda)_{\ominus} - 2(\lambda)_{(2,2,1)}(\lambda)_{(2)} \\
&\quad - 2(\lambda)_{(2,2,1)}(\lambda)_{(2)} - 2(\lambda)_{(2,1)}(\lambda)_{(2,2)} \\
&\quad - 2(\lambda)_{(2,2,2)}(\lambda)_{(1)} - 2(\lambda)_{(2,2)}(\lambda)_{(2,1)} \\
&\quad - 2(\lambda)_{(2,2)}(\lambda)_{(2,1)} - 2(\lambda)_{(2)}(\lambda)_{(2,2,1)} = 12, \\
(\lambda)_{(2,2,1,2,1)} &= (f^1(\lambda))_{(2,2,1,2)} \\
&= -2(\lambda)_{(2,2,1,2)}(\lambda)_{\ominus} - 2(\lambda)_{(2,1,2)}(\lambda)_{(2)} \\
&\quad - 2(\lambda)_{(2,2,2)}(\lambda)_{(1)} - 2(\lambda)_{(2,2)}(\lambda)_{(2,1)} \\
&\quad - 2(\lambda)_{(2,2,1)}(\lambda)_{(2)} - 2(\lambda)_{(2,1)}(\lambda)_{(2,2)} \\
&\quad - 2(\lambda)_{(2,2)}(\lambda)_{(1,2)} - 2(\lambda)_{(2)}(\lambda)_{(2,1,2)} = 4.
\end{aligned}$$

We get the same coefficients as in example 2.14, but this time we did not differentiate the vector fields manually for their computation.

### 3.2 Enclosure for the Remainder

After we have computed the coefficients of the solution's Fliess expansion up to the order  $p$ , we are now interested in the remainder term.

$$R_{\lambda(\tau, x_0)}^{p+1}(t, u) = \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b \left[ \mathcal{L}^b x \right] (t, u).$$

The analogue to Lohner's strategy [26] is to evaluate the Lie-derivatives in the remainder terms with the Mean Value Theorem from analysis at some intermediate values  $\xi_b \in [\tau, t]$ ,  $b \in \mathcal{M}_m^{p+1}$ , and get iterated integrals with constants as integrand. But in general the iterated integrals include non-continuous control functions. So the Mean Value Theorem cannot be applied. To get out of this difficulty we restrict the control functions to bounded functions.

**Definition 3.4 (control range)** *The vector  $\Delta = (\Delta^1, \dots, \Delta^m) \in \mathbb{I}^m$  gives the possible range for each control function, that is for all  $\alpha \in \{1, \dots, m\}$  and  $t \in [\tau, T]$  we have  $u^\alpha(t) \in \Delta^\alpha$ . For a multi-index  $b =$*

$(\beta_{|b|}, \dots, \beta_1) \in \mathcal{M}_m$  we define the product using interval arithmetics

$$\Delta^b = \prod_{k=1}^{|b|} \Delta^{\beta_k},$$

where  $\Delta^0 := [1, 1] = [1] = 1$ . For the empty index we define  $\Delta^\ominus := 1$ .

For the rest of this thesis we assume the control functions to be bounded and the control range to be known.

This restriction is not very serious as the following simple example shows. We consider a one-dimensional linear control system  $\dot{x} = u$ , where  $u$  is an unbounded control function, with the initial condition  $x(0) = 0$ . Then we can reach any point  $y \in \mathbb{R}$  for an arbitrary small time  $t > 0$  with the constant control function  $u \equiv \frac{y}{t}$ , thus it makes no sense to compute an enclosure for the unbounded set  $\{\lambda(t, 0, 0, u) | u \in \mathcal{U}\}$  numerically. Hence we need to confine the control functions with known control ranges.

Assuming bounded control functions we can formulate at least the following theorem.

**Theorem 3.5 (enclosure for the remainder)** *Let  $G : [\tau, T] \times \mathcal{U}^m \rightarrow \mathbb{R}^n$  be a measurable function and  $b \in \mathcal{M}_m$  a multi-index. Then the iterated integral can be enclosed by*

$$\mathcal{I}_b[G](t, u) \in G([\tau, t], u) \Delta^b \frac{(t - \tau)^{|b|}}{|b|!}.$$

*The remainder term of the solution's Fliess-expansion of order  $p$  can be enclosed with*

$$R_x^{p+1}(t, u) \in \frac{(t - \tau)^{p+1}}{(p + 1)!} \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{L}^b x([\tau, t]) \Delta^b.$$

We remember  $\mathcal{L}^b x(s) = \mathcal{L}^b x(s, \lambda(s, \tau, x_0, u))$  is the evaluation of the  $b$ -th Lie derivative of  $F(t, x) = x$  at  $(s, \lambda(s, \tau, x_0, u))$ .

**Proof:** For  $j = 1, \dots, |b|$  there exist numbers  $\beta_j \in \{1, \dots, m\}$  with  $b = (\beta_{|b|}, \dots, \beta_1)$ . We construct the enclosure for the iterated integral starting

from the inner integral. With  $s_j \in [\tau, t]$  follows:

$$\begin{aligned}
G(s_j, u) &\in G([\tau, t], u) \\
G(s_j, u) \prod_{j=1}^{|b|} u^{\beta_j}(s_j) &\in G([\tau, t], u) \Delta^b \text{ for all } s_j \in [\tau, t] \\
\int_{\tau}^{s_{j-1}} G(s_j, u) \prod_{j=1}^{|b|} u^{\beta_j}(s_j) ds_j &\in G([\tau, t], u) \Delta^b (s_{j-1} - \tau) \\
\mathcal{I}_b[G](t, u) &\in G([\tau, t], u) \Delta^b \frac{(t - \tau)^{|b|}}{|b|!}.
\end{aligned}$$

Then for the remainder of a solution  $x$  follows:

$$\begin{aligned}
R_x^{p+1}(t, u) &= \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{I}_b[\mathcal{L}^b x](t, u) \\
&\in \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{L}^b x([\tau, t]) \Delta^b \frac{(t - \tau)^{|b|}}{|b|!} \\
&\in \frac{(t - \tau)^{p+1}}{(p+1)!} \sum_{b \in \mathcal{M}_m^{p+1}} \mathcal{L}^b x([\tau, t]) \Delta^b.
\end{aligned}$$

■

Clearly we cannot compute the integral kernels of the remainder term  $\mathcal{L}^b x([\tau, t])$ ,  $b \in \mathcal{M}_m^{p+1}$ , explicitly. But we can compute a superset with the same recursion formula as for the coefficients of the solution's Fliess-expansion (see algorithm 3.2). Therefore we formulate a lemma to compute the Lie derivative  $\mathcal{L}^b x(s, x_s)$  at a given time  $s \in [\tau, T]$  and state space vector  $x_s \in \mathbb{R}^n$ . First of all  $s$  and  $x_s$  are independent.

**Lemma 3.6 (Lie-derivatives at arbitrary times)** *Let  $s \in [\tau, T]$  be an arbitrary time and  $x_s \in \mathbb{R}^n$ . We define recursively for all  $b \in \mathcal{M}_m$ , with  $|b| \leq p$ , and  $\beta \in \{0, \dots, m\}$  the coefficients:*

$$\begin{aligned}
(\lambda(s, x_s))_{\odot} &:= x_s, \\
(\lambda(s, x_s))_{(b, \beta)} &:= \begin{cases} 0, & \text{for } \beta = 0 \\ (f^{\beta}(\cdot, \lambda(s, x_s)))_b, & \text{for } \beta \in \{1, \dots, m\}. \end{cases}
\end{aligned}$$

*Then the Lie-derivative at  $(s, x_s)$  coincides with the appropriate coefficient.*

$$\mathcal{L}^b x(s, x_s) = (\lambda(s, x_s))_b, \text{ for all } b \in \mathcal{M}_m, |b| \leq p+1.$$

**Proof:** The idea of the proof is to develop a Fliess-expansion for  $\lambda(\cdot, s, x_s, u)$  at the time  $s$ . Therefore we need a more general definition for the iterated integrals than in definition 2.4. For a causal function  $G$ , indices  $\alpha \in \{0, \dots, m\}$  and  $b \in \mathcal{M}_m$  we define:

$$\begin{aligned} {}_s\mathcal{I}_\ominus[G](t, u) &:= G(t, u) \\ {}_s\mathcal{I}_{(\alpha, b)}[G](t, u) &:= {}_s\mathcal{I}_b \left[ \int_\tau^\cdot G(s, u) u^\alpha(s) \, ds \right] (t, u). \end{aligned}$$

So we can define the Fliess-expansion at the initial condition  $(s, x_s)$  of order  $p+1$  similar to definition 2.16:

$$\lambda(t, s, x_s, u) = \sum_{i=0}^{p+1} \sum_{b \in \mathcal{M}_m} (\lambda(s, x_s))_b {}_s\mathcal{I}_b[G](t, u) + {}_sR_{\lambda(s, x_s)}^{p+2}(t, u).$$

With  ${}_sR_{\lambda(s, x_s)}^{p+2}$  we denote the corresponding remainder term. Because we use the same recursion as in theorem 3.1 we get a Fliess-expansion for the solution. On the other hand we reformulate theorem 2.12 and its proof with  $s$  instead of  $\tau$  and with the uniqueness of the coefficients of Fliess-expansions (see theorem 2.28 for  $s$  but  $\tau$ ) we compare the coefficients and get the assertion:

$$\mathcal{L}^b x(s, x_s) = (\lambda(s, x_s))_b, \text{ for all } b \in \mathcal{M}_m, |b| \leq p+1.$$

■

We are interested at the value of the Lie-derivative at the whole time interval  $\mathcal{L}^b x([\tau, T])$ . For the present we assume to know a “rough” enclosure of the solution evaluated at all times  $x([\tau, T]) = \lambda([\tau, T], \tau, x_0, u)$ . Then lemma 3.6 gives recursion formulas to compute the Lie-derivatives evaluated at all times  $[\tau, T]$  and we are ready to formulate the algorithm.

**Algorithm 3.7 (enclosures for the remainder terms)**

**Step 0:** We start with an interval  $R^\ominus \supseteq x([\tau, t])$ , which includes all solutions.

**Step  $i$ , for  $i = 1, \dots, p+1$ :** We compute recursively for all  $b \in \mathcal{M}_m^{i-1}$  and all  $\alpha \in \{1, \dots, m\}$  the coefficients  $R^{(b, \alpha)} \supseteq (f^\alpha(\cdot, R))_b$  from  $R^c$ ,  $|c| \leq i$ , where  $R$  is the Fliess-expansion composed by the coefficients  $R^c$ .



Then we get enclosures  $R^b \supset \mathcal{L}^b x([\tau, t])$ , for  $b \in \mathcal{M}_m^{p+1}$ . With theorem 3.5 we enclose the entire remainder.

$$R_x^{p+1}(t, u) \in \frac{(t - \tau)^{p+1}}{(p + 1)!} \sum_{b \in \mathcal{M}_m^{p+1}} R^b \Delta^b.$$

In general the superset for  $x([\tau, t])$  is not known. We will discuss in section 3.3 a strategy how to get an initial enclosure for all solutions of an initial interval problem with bounded controls. At least we will give a criterion to identify an interval as enclosure for all solutions.

### 3.3 Initial Enclosure

To compute an enclosure for the remainder term  $R_{\lambda(\tau, [x_0])}^{p+1}(t, u)$  we need a rough initial enclosure for all solutions  $\lambda([\tau, T], [x_0])$ . We will not develop a strategy which finds an initial enclosure in any case, but we will give a sufficient condition for identifying a given interval as an initial enclosure.

**Theorem 3.8 (initial enclosure)** *Consider an interval  $[\hat{x}] \in \mathbb{R}^n$ , which fulfils the inclusion:*

$$x_0 + [0, T - \tau] \sum_{\alpha=1}^m f^\alpha([\tau, T], [\hat{x}]) \Delta^\alpha \subseteq [\hat{x}].$$

*Then the set of continuous functions  $C([\tau, T], [\hat{x}])$  with range  $[\hat{x}]$  contains the unique solution  $\lambda(\cdot, \tau, x_0, u)$ .*

**Proof:** The set  $C([\tau, T], [\hat{x}])$  is a closed subset of  $C([\tau, T], \mathbb{R}^n)$ . We show that the operator  $W$  from theorem 1.3 maps  $C([\tau, T], [\hat{x}])$  on itself. Then from Banach fixed point theorem follows the existence of the fixed point and therewith the existence of the solution in  $[\hat{x}]$ .

$$\begin{aligned}
W[C([\tau, T], [\hat{x}])](t) &\subseteq x_0 + \int_{\tau}^t \sum_{\alpha=1}^m f^{\alpha}(s, [\hat{x}]) u^{\alpha}(s) \, ds \\
&\subseteq x_0 + \int_{\tau}^t ds \sum_{\alpha=1}^m f^{\alpha}([\tau, T], [\hat{x}]) \Delta^{\alpha} \\
&\subseteq x_0 + [0, T - \tau] \sum_{\alpha=1}^m f^{\alpha}([\tau, T], [\hat{x}]) \Delta^{\alpha} \\
&\subseteq [\hat{x}]
\end{aligned}$$

So  $W$  is a contraction on  $C([\tau, T], [\hat{x}])$ . ■

To verify an initial enclosure for  $\lambda(\cdot, \tau, x_0, u)$ , with an initial interval instead of an initial value, we extend theorem 3.8 to its union over all initial values  $x_0 \in [x_0]$ :

$$[x_0] + [0, T - \tau] \sum_{\alpha=1}^m f^{\alpha}([\tau, T], [\hat{x}]) \Delta^{\alpha} \subseteq [\hat{x}]. \quad (3.3)$$

As a strategy Lohner [26] put forward to start with the initial interval  $[\hat{x}] := [x_0]$  and inflate it by a small  $\varepsilon > 0$  (see definition 1.9) recursively until equation 3.3 is satisfied. If the time  $T$  is small enough, an initial enclosure can be found in any case. Thereby lemma 1.10 assures  $[x_0] \subset [\hat{x}]$ .

In numerical examples it turned out it is good to start with an interval, which is computed by the recursion of theorem 3.1 but with initial value  $([\hat{x}])_{\ominus} := [x_0]$ . The Fliess-expansion defined by this interval coefficients can be evaluated for the whole time interval  $[\tau, T]$  and control range:

$$[\hat{x}] := \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} ([\hat{x}])_b \mathcal{I}_b([\tau, T], \Delta^b). \quad (3.4)$$

Note that this is not a rigorous enclosure for  $\lambda([\tau, T], \tau, [x_0], u)$ , because the remainder term was neglected. But due the hope of small remainders we can inflate  $[\hat{x}]$  by  $\varepsilon > 0$  like in the paragraph above, until we can verify the enclosure.

### 3.4 Error Propagation

In this section we interpret the initial interval  $[x_0]$  as an error bound for the initial value  $x_0 + z$ , with  $z \in [x_0] - x_0$  and linearise the coefficients

of the solution's Fliess-expansion at  $x_0$ . For each  $b \in \mathcal{M}_m$  there exists a  $\xi_b \in [x_0, z]$ , such that

$$(\lambda(\tau, x_0 + z))_b = (\lambda(\tau, x_0))_b + \frac{\partial}{\partial x_0} (\lambda(\tau, \xi_b))_b * z.$$

The solution  $\lambda$  depends on the nonsmooth control functions  $u^\alpha$ , for  $\alpha \in \{1, \dots, m\}$ , so we cannot expect  $\lambda$  to be differentiable in the initial value  $x_0$ . But we assumed the vector fields  $f^\alpha$ ,  $\alpha \in \{1, \dots, m\}$ , to be smooth enough. The coefficients  $(\lambda(\tau, x_0))_b = \mathcal{L}^b x(\tau, x_0)$ ,  $|b| \leq p$ , only depend on the right hand side vector fields but not on the control functions. So the existence of  $\frac{\partial}{\partial x_0} (\lambda(\tau, x_0))_b$  is part of our basic assumption.

**Definition 3.9 (error propagation matrix)** *We call the matrix valued Fliess-expansion,*

$$A(t, u) := \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} \frac{\partial}{\partial x_0} (\lambda(\tau, x_0))_b \mathcal{I}_b(t, u),$$

*the error propagation matrix for the solution  $\lambda(\tau, x_0)$ , where the coefficients  $(A(\tau, x_0))_b := \frac{\partial}{\partial x_0} (\lambda(\tau, x_0))_b$  depend on the initial conditions.*

We can give a recursion formula for the computation of the coefficients of the error propagation matrix.

**Lemma 3.10 (recursion for the error propagation)** *The coefficients of the Fliess-expansion of the error propagation matrix satisfy the following recursion formula:*

$$\begin{aligned} (A(\tau, x_0))_\ominus &= I, \\ (A(\tau, x_0))_{(b, \alpha)} &= \begin{cases} 0, & \alpha = 0, \\ \left( \frac{\partial}{\partial x} f^\alpha(\cdot, \lambda(\tau, x_0)) A(\tau, x_0) \right)_b, & \text{otherwise,} \end{cases} \end{aligned}$$

*where  $I$  is the identity matrix of dimension  $n$ .*

**Proof:** Because  $\lambda(\tau, x_0)$  is not differentiable in  $x_0$  generally, we define the corresponding function  $\bar{\lambda}(\tau, x_0)$ , which is its Fliess-expansion without remainder,

$$\bar{\lambda}(\tau, x_0) := \sum_{i=0}^p \sum_{b \in \mathcal{M}_m^i} (\lambda(\tau, x_0))_b,$$

which is differentiable in  $x_0$ . Then for each index  $\alpha \in \{1, \dots, m\}$  the function  $f^\alpha(\bar{\lambda}(\tau, x_0))$  is differentiable in  $x_0$ , because  $f^\alpha$  is assumed to be smooth enough. For this proof we skip the first argument of  $f^\alpha$ :

$$\frac{\partial}{\partial x_0} f^\alpha(\bar{\lambda}(\tau, x_0)) = \frac{\partial}{\partial x} f^\alpha(\bar{\lambda}(\tau, x_0)) \frac{\partial}{\partial x_0} \bar{\lambda}(\tau, x_0).$$

Integration and differentiation in different variables can be exchanged. So the differentiation can be executed inside of every iterated integral in the remainder term, which leads to

$$\frac{\partial}{\partial x_0} R_{f^\alpha(\bar{\lambda}(\tau, x_0))}^{p+1}(t, u) = R_{\frac{\partial}{\partial x_0} f^\alpha(\bar{\lambda}(\tau, x_0))}^{p+1}(t, u),$$

and then with the uniqueness of the coefficients of Fliess-expansions to

$$\frac{\partial}{\partial x_0} (f^\alpha(\bar{\lambda}(\tau, x_0)))_b = \left( \frac{\partial}{\partial x_0} f^\alpha(\bar{\lambda}(\tau, x_0)) \right)_b.$$

Then we get for  $\alpha \in \{1, \dots, m\}$  and  $b \in \mathcal{M}_m$ ,  $|b| < p$  with theorem 3.1

$$\begin{aligned} (A)_\ominus &= \frac{\partial}{\partial x_0} (\lambda(\tau, x_0))_\ominus = I \\ (A)_{(b, 0)} &= \frac{\partial}{\partial x_0} (\lambda(\tau, x_0))_{(b, 0)} = 0 \\ (A)_{(b, \alpha)} &= \frac{\partial}{\partial x_0} (\lambda(\tau, x_0))_{(b, \alpha)} = \frac{\partial}{\partial x_0} (f^\alpha \lambda(\tau, x_0))_b \\ &= \frac{\partial}{\partial x_0} (f^\alpha \bar{\lambda}(\tau, x_0))_b = \left( \frac{\partial}{\partial x_0} f^\alpha(\bar{\lambda}(\tau, x_0)) \right)_b \\ &= \left( \frac{\partial}{\partial x} f^\alpha(\bar{\lambda}(\tau, x_0)) \frac{\partial}{\partial x_0} \bar{\lambda}(\tau, x_0) \right)_b \\ &= \left( \frac{\partial}{\partial x} f^\alpha(\lambda(\tau, x_0)) \frac{\partial}{\partial x_0} \lambda(\tau, x_0) \right)_b \\ &= \left( \frac{\partial}{\partial x} f^\alpha(\bar{\lambda}(\tau, x_0)) A \right)_b \end{aligned}$$

■

**Remark 3.11** The linearisation of the error  $\frac{\partial}{\partial x_0} \lambda(\tau, x_0)$  acts like the Fliess-expansion computed by algorithm 3.2 for the solution of the matrix-valued variational equation,

$$\dot{A}(t) = \sum_{\alpha=1}^m g^\alpha(t, A(t)) u^\alpha(t), \quad A(\tau) = I,$$

$$\text{with } g^\alpha(t, A(t)) := \frac{\partial}{\partial x} f^\alpha(t, \lambda(\tau, x_0)) A(t),$$

without remainder term.

**Algorithm 3.12 (error propagation matrix)**

**Step 0:** Provide a Fliess-expansion with interval valued coefficients for all solutions with initial condition  $x(\tau) = [x_0]$  and insert it into the right hand side derivatives  $\frac{\partial}{\partial x} f^\alpha(\cdot, x)$ , for  $\alpha \in \{1, \dots, m\}$ . We start the computation with the identity matrix as initial value  $(A)_\ominus = I$ .

**Step  $i$ , for  $i = 1, \dots, p$ :** We compute recursively for all  $b \in \mathcal{M}_m^i$  the coefficients  $(z)_b$  of the Fliess-expansion with lemma 3.10.

As result of algorithm 3.12 we get the coefficients of the error propagation matrix  $A$ , which include the derivative of the coefficients of the solution evaluated at the arbitrary (and unknown) intermediate value  $\xi_b \in [x_0]$ :

$$\frac{\partial}{\partial x_0} (\lambda(\tau, \xi_b))_b \in (A)_b,$$

for all  $b \in \mathcal{M}_m$ ,  $|b| \leq p$ .

### 3.5 The Algorithm

Now we have developed all necessary tools to enclose all solutions of the control affine system (1.1) with the initial interval condition

$$\dot{x}(t) = \sum_{\alpha=1}^m f^\alpha(t, x) u^\alpha(t), \quad x(\tau) \in [x_0].$$

The control functions are assumed to be bounded, that is for every  $t \in [\tau, T]$  and  $\alpha \in \{1, \dots, m\}$  we have  $u^\alpha(t) \in \Delta^\alpha$ . The right hand side vector fields are continuously differentiable. This assures the Lipschitz-condition in  $x$  and the existence and uniqueness of the solution for every choice of initial value and control function. Furthermore the right hand side vector fields are required to be composed by operations, which are defined for Fliess-expansions.

**Algorithm 3.13 (enclosure for all solutions)** We compute an enclosure of order  $p \in \mathbb{N}$  for all solutions at time  $T$  with the following steps.

**Step 1:** Choose first an initial value from the initial interval  $x_0 \in [x_0]$ , for example the centre of  $[x_0]$ , and compute the coefficients  $(\lambda(\tau, x_0))_b$ , for all  $|b| \leq p$ , with algorithm 3.2 recursively.

**Step 2:** Compute with algorithm 3.2 a “rough” initial enclosure for the coefficients of the initial interval problem  $(\lambda(\tau, [x_0]))_b$ , for all  $|b| \leq p$ , starting with  $(\lambda(\tau, [x_0]))_\ominus = [x_0]$ .

**Step 3:** Evaluate the Fliess-expansion for  $\lambda(\tau, [x_0])$  at the whole time interval  $[\tau, T]$  and for the control ranges  $\Delta^\alpha$ ,  $\alpha \in \{1, \dots, m\}$  (use theorem 3.5). Neglect the remainder and check if for the resulting interval the initial enclosure condition 3.3 holds true. If not, inflate the interval by a small  $\varepsilon > 0$  componentwise (cp. definition 1.9) and check again till the enclosure condition is satisfied.

**Step 4:** Compute an enclosure for the remainder term  $R_{\lambda(\tau, [x_0])}^{p+1}(t, \Delta)$  with algorithm 3.7. Use the interval computed in step 3 as an enclosure for the initial set  $R^\ominus \supseteq \lambda([\tau, T], \tau, [x_0], \Delta)$ .

**Step 5:** Compute the Fliess expansion of the error propagation matrix  $A$  with algorithm 3.12. Insert thereby the “rough” enclosure of the solution’s Fliess-expansion from step 2 into  $\frac{\partial}{\partial x} f^\alpha(\cdot, \lambda(\tau, x_0))$ .

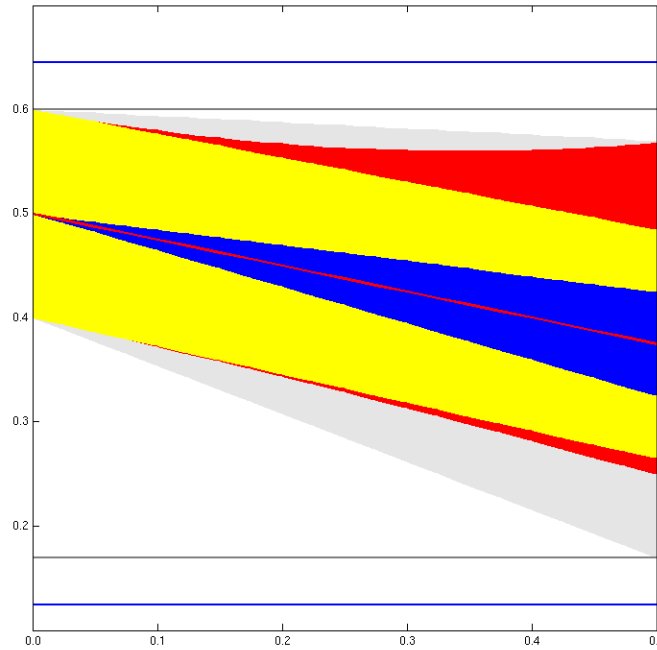
**Step 6:** Evaluate the Fliess-expansion of  $\lambda(\tau, x_0)$  and the Fliess-expansion of the error propagation matrix  $A$  multiplied with the error vector  $[x_0] - x_0$  at time  $T$ . Add them up together with the enclosure for the remainder from step 4 at time  $T$ .

The evaluations in step 6 can be done for any time  $t \in [\tau, T]$ , if dense output is part of the task.

In figure 3.1 the different steps of algorithm 3.13 are illustrated for the control system we used already in the examples 2.14 and 3.3,

$$\dot{x} = u - x^2, \quad (3.5)$$

with the initial interval condition  $x(0) \in [x_0] := [0.4, 0.6]$  and the control range  $\Delta := [-0.1, 0.1]$ . The order was chosen with  $p = 1$ . The Fliess-expansion for  $\lambda(\tau, x_0)$  from step 1 is drawn with the centred red line for  $u \equiv 0$ . For the whole control range  $\Delta$  it swells to the blue area. In the yellow area the propagation of the initial interval by the error propagation



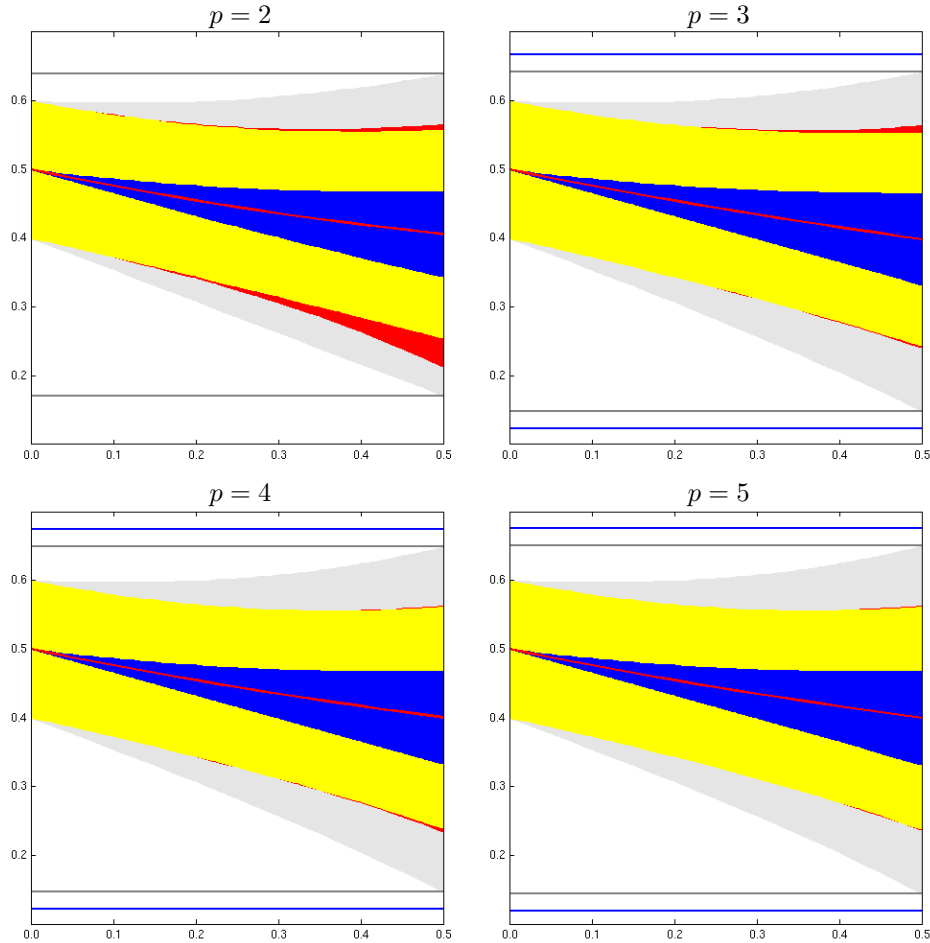
**Figure 3.1:** Enclosure for all solutions of  $\dot{x} = u - x^2$ ,  $x(0) \in [0.4, 0.6]$  and  $\Delta := [-0.1, 0.1]$  computed for order  $p = 1$ .

matrix is added (step 5). The enclosure of the remainder term (step 4) is represented by the red area. So the enclosure for all solutions is given by the union of the blue, yellow and red region.

The grey area in the background shows the evaluation of the solution's Fliess-expansion with interval coefficients from step 1 and the grey lines show its lower and upper bounds. They are inflated in step 2 to the rough initial enclosures (outer blue lines).

Figure 3.2 illustrates the same sets for the orders  $p = 2, 3, 4, 5$ . We can see the remainder enclosure shrinking with the order, but the resulting enclosures seem to be the same in every picture.

For a better ability to compare table 3.1 gives the enclosures for all solutions at time  $T$  for the orders  $p = 1, \dots, 6$ . We remember our algorithm gives guaranteed bounds, that is we know for sure that all solutions are contained in each of the resulting intervals. So we can take the intersection of all computed intervals and get again an enclosure for all solutions (see the line denoted with “1 – 6”). This enclosure is obviously better than all the others, because they are bigger but do not contain more information. Thus they overestimate. In addition we can see that the results do not get



**Figure 3.2:** Enclosure for all solutions of  $\dot{x} = u - x^2$ ,  $x(0) \in [0.4, 0.6]$  and  $\Delta := [-0.1, 0.1]$  computed for order  $p = 2, 3, 4, 5$ .

smaller with higher order.

In the one-dimensional case there is another way to find an enclosure for all solutions. We can compute the solution for the maximal initial value together with the maximal control function with our algorithm. Because we have chosen a concrete initial value and control function we compute an enclosure for the unique solution. The same we do for the minimal initial value together with the minimal control function:

$$\begin{aligned} \lambda(0.5, 0.0, 0.6, u \equiv 0.1) &\in [0.499850, 0.501708], \\ \lambda(0.5, 0.0, 0.4, u \equiv -0.1) &\in [0.290815, 0.291121]. \end{aligned}$$

From control theory we know that every other solution of the initial in-



$p$	result of algorithm 3.13 at $T = 0.5$
1	[0.249358, 0.568260]
2	[0.212309, 0.567217]
3	[0.239688, 0.565434]
4	[0.233871, 0.563961]
5	[0.236318, 0.564492]
6	[0.235529, 0.564177]
1-6	[0.249358, 0.563961]
extremal controls	[0.290815, 0.501708]

**Table 3.1:** Enclosure for all solutions of  $\dot{x} = u - x^2$ ,  $x(0) \in [0.4, 0.6]$  and  $\Delta := [-0.1, 0.1]$  computed for different orders  $p$  at time  $T = 0.5$ , their intersection and a verified result for extremal controls.

terval problem with our previously used control range is in between this two solutions. The convex hull of the extremal enclosures gives a much better rigorous result than we got with our algorithm. In table 3.1 it is denoted with “extremal controls”. For more details of this strategy for the one-dimensional systems see [4, 5].

Obviously our algorithm overestimates the resulting interval. Responsible is the enclosure of the control functions by the control range. To find narrow enclosures is still an open problem.

The implementation of algorithm 3.13 was done in C++ . An object oriented programming language, like C++ , lend itself to define type classes for Fliess-expansions and formulas composed from basic arithmetic operations. Then the Fliess-expansion can be, roughly speaking, inserted into the formula class and gives as result again a Fliess-expansion, if all used arithmetic operations are defined for it. The multi-indices are represented as serial numbers, so the coefficients are linear sequences and can be handled like vectors. The library C-XSC was used for interval arithmetic.

The numerical effort especially for the product formula (cp. theorem 2.33) is very high. For the order 5 and 6 the computation on a Linux PC with a Pentium 4, 2.80 GHz, processor took already several minutes.



# Chapter 4

## Application

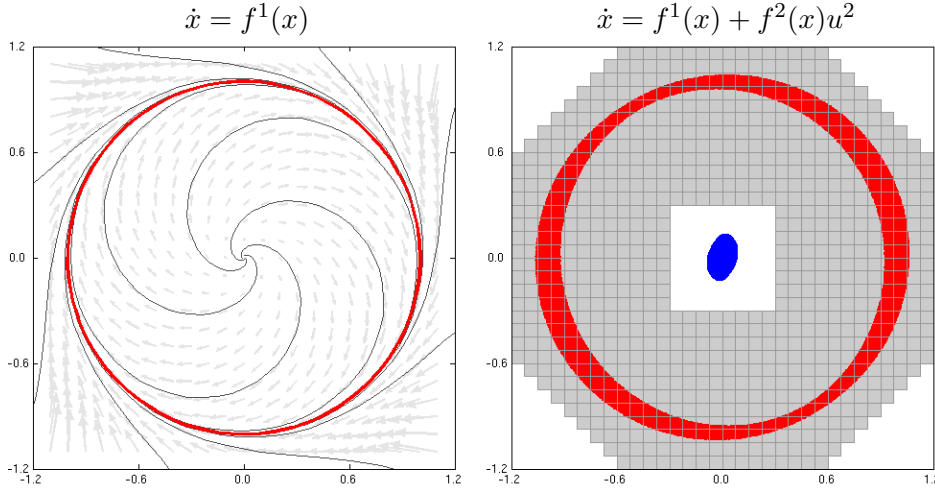
In this chapter we will show how our new algorithm can be used to compute practical problems. We take the 2-dimensional system from Aulbach [2], equation (2.19), where the unit circle is an attractive periodic solution and the origin is an unstable fixed point:

$$\dot{x} = f^1(x), \text{ with } f^1(x) := \begin{pmatrix} x_2 \\ -x_1 \end{pmatrix} + \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} (1 - (x_1)^2 - (x_2)^2). \quad (4.1)$$

The state variable  $x \in \mathbb{R}^2$  has the components  $x_1$  and  $x_2$ . The phase portrait is given in figure 4.1(left). We define the constant vector field  $f^2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ . With the control functions  $u^1 \equiv 1$  and  $u^2 : \mathbb{R} \rightarrow \Delta^2$  we get a control affine system in the notation of definition 1.1. We choose  $\Delta^1 := [1]$  and  $\Delta^2 := [-0.1, 0.1]$  as the control ranges. The parameters are the dimension of the state space  $n = 2$  and the dimension of the control  $m = 2$ . The control system is defined for all times in  $\mathbb{R}$ :

$$\dot{x} = \begin{pmatrix} x_2 \\ -x_1 \end{pmatrix} + \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} (1 - (x_1)^2 - (x_2)^2) + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u^2. \quad (4.2)$$

This system was analysed by Kowalski and Stender in [23]. Amongst others they computed an invariant ring-shaped set including the unit circle (cp. figure 4.1, the red set on the right hand side picture). Depending on the point of view there are different names denoting this set. For instance it is the reachability set of the unit circle. With definitions of Colonius and



**Figure 4.1:** The phase portrait of the attractive unit circle system (left). All trajectories are heading clockwise to the unit circle. On the right hand side we see the variant control set (blue) around the origin and the invariant control set (red) including the unit circle for  $u^2(s) \in \Delta^2 = [-\frac{1}{10}, \frac{1}{10}]$ .

Kliemann [5] it is an invariant control set, which is a subset of the state space where total controllability holds, i.e. every point can be reached from every other point at least approximately. In viability theory this set is the viability kernel of any of its supersets excluding the origin. We use the so called subdivision algorithm from Szolnoki [32, 33] to compute the viability kernel of the domain  $D$ . We define  $D$  as the union of all grey boxes in figure 4.1(right hand side picture). The set of all trajectories which stay in  $D$  for all times is

$$\text{Viab}(D) := \{x_0 \in D \mid \exists u \in \mathcal{U}^m \text{ with } \lambda(t, x_0, u) \in D, \forall t \in \mathbb{R}\}.$$

Therefore the system has to be discretised in time. For any  $h \geq 0$  we can compute a discrete viability kernel  $\text{Viab}_h(D)$  for time discrete system:

$$x_{i+1} = \lambda(h, x_i, u), \quad i \in \mathbb{N}.$$

Szolnoki proved in [32, prop. 3.5] that the discrete and the continuous viability kernels coincide if they are “deep inside” the domain. More precisely follows from  $\text{dist}(\text{Viab}_h(D), \mathbb{R}^n \setminus D) \leq \frac{1}{2}Kh$ , that  $\text{Viab}(D) = \text{Viab}_h(D)$ . Thereby  $K$  is an upper bound for the control system’s right hand side, i.e.  $\|\sum_{\alpha=1}^m f^\alpha(x)u^\alpha(t)\| \leq K$ . We use the numerical software INTLAB [31] to compute a superset for the right hand side’s length.

$$\|f^1(D) + f^2(D)\Delta^2\| \subset [0.3454, 1.8090]$$

This interval enclosure is guaranteed. We choose the upper bound  $K := 1.81$  and the time step  $h := 0.1$ . With these values we get a distance of at least 0.0905 we need after the computation between the discrete viability kernel and the boundary of the domain  $D$  to get the continuous viability kernel. One can see easily that it is fulfilled for the approximation in the picture. (The diagonal length of one grey box is approximately 0.1.)

The subdivision algorithm consists of the iteration of two steps. It starts with a collection of intervals in the state space and alternates a subdivision step with a selection step. The subdivision step bisects every interval into two parts, where the bisection axis, which is parallel, changes periodically. It creates a “finer” collection, that is the maximum diameter of the interval gets smaller. In the selection step a selection criterion decides which intervals are kept and which ones are removed. On the remaining collection of intervals again the subdivision step is applied, and so on. In this way one obtains a sequence of successively finer collections  $(\mathcal{C}_k)_{k \in \mathbb{N}_0}$ .

For finding the viability kernel we use the following selection step. We compute for every interval  $\mathcal{B}$  in the given collection  $\mathcal{C}_i$  the image of the map for the time discrete system

$$\Phi(\mathcal{B}) := \{\lambda(h, \mathcal{B}, u) | u \in \mathcal{U}^m, u([0, h]) \in \Delta\}.$$

We delete every interval which does not intersect with the image of the collection  $\Phi(\mathcal{C}_i) := \bigcup_{\mathcal{B} \in \mathcal{C}_i} \Phi(\mathcal{B})$ . The remaining collection  $\{\mathcal{B} \in \mathcal{C}_i | \mathcal{B} \cap \Phi(\mathcal{C}_i)\}$  we call again  $\mathcal{C}_i$  and repeat the selection step until it converges to the solution of the set valued equation:

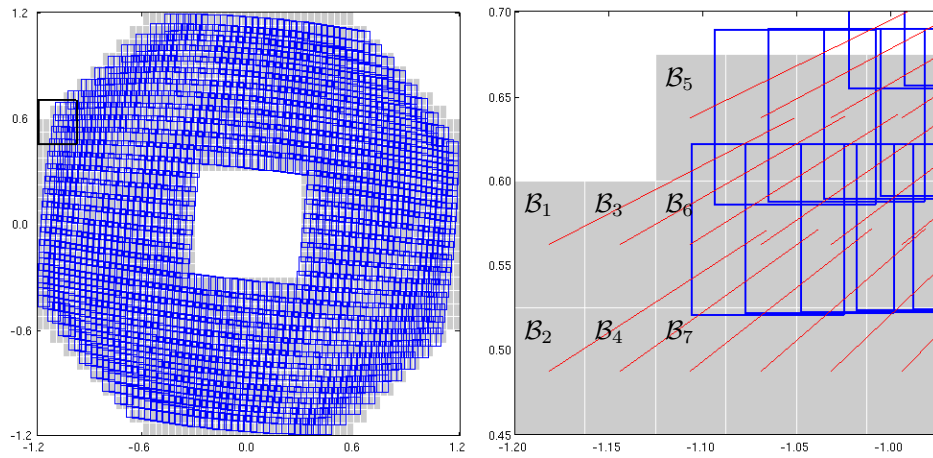
$$\mathcal{C}_i = \{\mathcal{B} \in \mathcal{C}_i | \mathcal{B} \cap \Phi(\mathcal{C}_i) \neq \emptyset\}.$$

The convergence is assured by the finite number of intervals in collection  $\mathcal{C}_i$ .

The sequence of collections  $(\mathcal{C}_i)_{i \in \mathbb{N}_0}$  converges from outside to the viability kernel of the time discrete system

$$\text{Viab}_h(\mathcal{UC}_0) = \bigcap_{i=0}^{\infty} \mathcal{UC}_i.$$

We denote with  $\mathcal{UC}_k$  the union of all intervals in the collection:  $\bigcup_{\mathcal{B} \in \mathcal{C}_k} \mathcal{B}$ . The subdivision algorithm was implemented by Dellnitz and Junge in GAIO [6] for differential equations. The extension to control systems was done by Szolnoki [32]. For both the computation of the images  $\Phi(\mathcal{B})$  is the crucial part of the algorithm. In praxis it turned out that it is sufficient to choose some test points in the interval  $\mathcal{B}$  and simulate for different control values



**Figure 4.2: left:** The grey boxes represent the initial collection  $\mathcal{C}_0$  and the blue boxes illustrate their images  $\Phi(\mathcal{B}), \mathcal{B} \in \mathcal{C}_0$ . **right:** The black box (left) is zoomed for a detailed view of the reachability relations. The images coming from intervals below are not drawn.

with standard Runge-Kutta algorithms. The number and distribution of test points and control values depends on the experience and the inventiveness of the user. For ODEs Junge [19] gave already a strategy to get out of the test point simulation from above a rigorous enclosure of  $\Phi(\mathcal{B})$  by the knowledge of a Lipschitz constant.

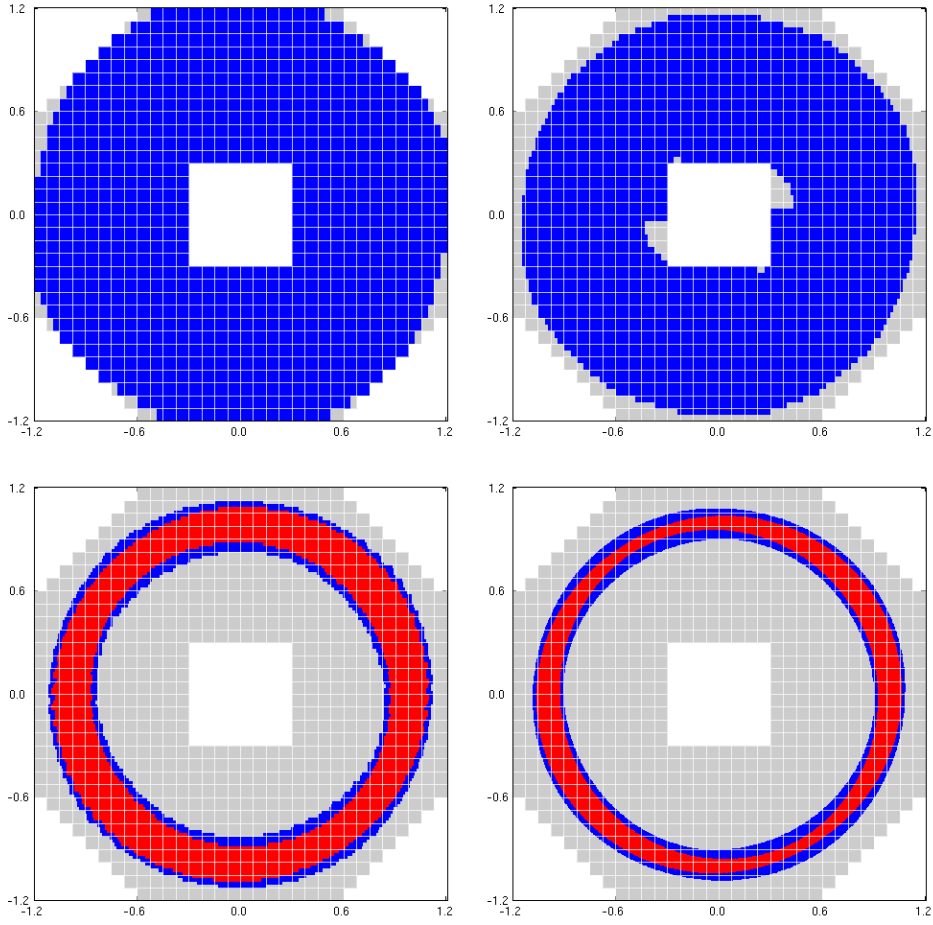
Algorithm 3.13 computes an enclosure for  $\Phi(\mathcal{B})$ . Thereby  $\mathcal{B}$  is the initial interval. The set  $\Phi(\mathcal{UC}_0)$  is shown in figure 4.2 as the union of the overlapping blue intervals. Each of them represents the image of one grey box. For each box the initial interval problem was solved with the order  $p = 2$ .

The right picture in figure 4.2 shows a detailed part of the left picture. The grey boxes are the initial intervals and the blue boxes are the enclosures for the time- $h$  maps. The red lines connect the centre points of the preimages with the centres of the corresponding image enclosures. We apply the selection step and need to delete the intervals  $\mathcal{B}_1, \dots, \mathcal{B}_4$ , because their intersection with the blue ranges are empty. By deleting  $\mathcal{B}_1$  and  $\mathcal{B}_2$  the intervals  $\mathcal{B}_5$  and  $\mathcal{B}_6$  get free, which means they have no intersection with the ranges of the remaining selection. Actually  $\mathcal{B}_7$  intersects the image of another interval outside of the shown section. This is the reason why it and its succeeding intervals are not deleted.

Figure 4.3 shows the covering of the viability kernel after 1, 3, 5 and 10 iterations. In the final picture we can see the distance to the boundary of

the initial set is big enough ( $> 0.0905$ ) to conclude from the discrete to the continuous viability kernel. The difference between the rigorously computed viability kernel (blue) and the one from test point simulation is based on the overestimating at adding up the control ranges. The wrapping effect has here no longer any effect, because the patching strategy (cp. section 1.2) works through the subdivision technique.

The picture for 10 iterations additionally shows the thinner set, which was computed with test point approximation (red). Like in example 3.5 the results were overestimated. But nevertheless this enclosure is verified. We proved numerically the existence of the continuous viability kernel inside the blue set. All computations were done with INTLAB , GAIO and C-XSC 2.0.



**Figure 4.3:** The cover for the viability kernel of the grey initial set after 1, 3, 5 and 10 iterations. The blue sets are computed with rigorous numerics, the red sets with test point approximation.



# Bibliography

- [1] J.-P. AUBIN, *Viability Theory*, Birkhäuser, Boston, Basel, Berlin, 1991.
- [2] B. AULBACH, *Gewöhnliche Differentialgleichungen*, Hochschultaschenbuch, Spektrum, Akad. Verlag, Heidelberg, Berlin, 1997.
- [3] F. COLONIUS AND T. KAPELA, *A Rigorous Numerical Algorithm for Controllability*. preprint, 2005. [\[WWW\]](#).
- [4] F. COLONIUS AND W. KLIEMANN, *Controllability and stabilization of one-dimensional systems near bifurcation points*, Systems & Control Letters, 24 (1995), pp. 87–95.
- [5] ———, *The Dynamics of Control*, Birkhäuser, Boston, 2000.
- [6] M. DELLNITZ, G. FROYLAND, AND O. JUNGE, *The algorithms behind GAIO – Set oriented numerical methods for dynamical systems*, in Ergodic Theory, Analysis, and Efficient Simulation of Dynamical Systems, B. Fiedler, ed., Springer, 2001, pp. 145–174. [\[WWW\]](#).
- [7] M. FLIESS, *Fonctionnelles causales non linéaires et indéterminées non commutatives*, Bulletin de la S.M.F., 109 (1981), pp. 3–40.
- [8] ———, *Réalisation locale des systèmes non linéaires, algèbres de lie filtrées transitives et séries génératrices non commutatives*, Inventiones Mathematicae, 71 (1983), pp. 521–537.
- [9] ———, *Vers une notion de dérivation fonctionnelle causale*, Annales de l’Institut Henri Poincaré - Analyse non linéaire, 3 (1986), pp. 67–76.
- [10] T. GAYER, *Controlled And Perturbed Systems Under Parameter Variation*, Shaker, Aachen, 2003. [\[WWW\]](#).
- [11] L. GRÜNE, *Asymptotic Behavior of Dynamical and Control Systems under Perturbation and Discretization*, vol. 1783 of Lecture Notes in Mathematics, Springer, Berlin Heidelberg New York, 1981.

- [12] L. GRÜNE AND P. E. KLOEDEN, *Higher order numerical schemes for affinely controlled nonlinear systems*, Numerische Mathematik, 89 (2001), pp. 669–690. [\[WWW\]](#).
- [13] G. I. HARGREAVES, *Interval analysis in MATLAB*, Numerical Analysis Report No. 416, Manchester Centre for Computational Mathematics, Manchester, England, Dec. 2002. [\[WWW\]](#).
- [14] D. HINRICHSSEN AND A. J. PRITCHARD, *Mathematical Systems Theory I*, Texts in Applied Mathematics, Springer-Verlag, Berlin and Heidelberg, 2005.
- [15] W. HOFSCHESTER AND W. KRÄMER, *C-XSC 2.0: A C++ library for extended scientific computing*, in Numerical Software with Result Verification, R. Alt, A. Frommer, R. Kearfott, and W. Luther, eds., vol. 2991 of Lecture Notes in Computer Science, Springer, Heidelberg, 2004, pp. 15–35. [\[WWW\]](#).
- [16] W. HOFSCHESTER, W. KRÄMER, S. WEDNER, AND A. WIETHOFF, *C-XSC 2.0 - a C++ class library for extended scientific computing*. Preprint 2001/1, Wissenschaftliches Rechnen / Softwaretechnologie, Universität Wuppertal, 2001. [\[WWW\]](#).
- [17] A. ISIDORI, *Nonlinear Control Systems*, Springer-Verlag, London, 3 ed., 1995.
- [18] J. JAULIN, M. KIEFFER, O. DIDRIT, AND E. WALTER, *Applied Interval Analysis*, Springer-Verlag, London, 2001.
- [19] O. JUNGE, *Rigorous discretization of subdivision techniques*, in EQUADIFF 99, Proceedings of the International Conference on Differential Equations, B. Fiedler, K. Gröger, and J. Sprekels, eds., Singapore, 2000, World Scientific, pp. 916–917.
- [20] T. KAPELA, *N-BODY choreographies with a reflectional symmetry - computer assisted existence proofs*, in EQUADIFF 2003, Proceedings of the International Conference on Differential Equations, F. Dumortier, H. Broer, J. Mawhin, A. Vanderbauwhede, and S. V. Lunel, eds., World Scientific, 2005, pp. 999–1005. [\[WWW\]](#).
- [21] T. KAPELA AND P. ZGLICZYNSKI, *The existence of simple choreographies for the n-body problem - a computer assisted proof*, Nonlinearity, 16 (2003), pp. 1899–1918. [\[WWW\]](#).

- [22] R. KLATTE, U. KULISCH, A. WIETHOFF, C. LAWOW, AND M. RAUCH, *A C++ Class Library for Extended Scientific Computing*, Springer, Heidelberg, New York, 1993.
- [23] J. KOWALSKI AND T. STENDER, *Numerisches Praktikum in Kontrolltheorie*, Augsburg, 2003.
- [24] R. KRAWCZYK, *Newton-Algorithmen zur Bestimmung von Nullstellen mit Fehlerschranken*, Computing, 4 (1969), pp. 187–201.
- [25] C. LESIAK AND A. KRENER, *The existence and uniqueness of Volterra series for nonlinear systems*, IEEE transactions on automatic control, 23 (1978), pp. 1090–1095.
- [26] R. LOHNER, *Einschließung der Lösung gewöhnlicher Anfangs- und Randwertaufgaben und Anwendungen*, Universität Karlsruhe, Karlsruhe, 1988.
- [27] —, *On the Ubiquity of the Wrapping Effect in the Computation of Error Bounds*, in Perspectives on Enclosure Methods, U. Kulisch, R. Lohner, and A. Facius, eds., Springer Verlag, Wien, New York, 2001, pp. 201–217.
- [28] A. MARQUARDT, *Subdivisionsalgorithmen für Kontrollierbarkeitsprobleme*. Diploma thesis, Universität Augsburg, 2001. [\[WWW\]](#).
- [29] R. E. MOORE, *Interval Arithmetic*, Prentice-Hall, Englewood Cliffs, NJ, 1966.
- [30] L. B. RALL, *Automatic differentiation: Techniques and applications*, Lecture Notes in Comput. Sci., Springer-Verlag, New York, 1981.
- [31] S. M. RUMP, *INTLAB - INTerval LABoratory*, in Developments in Reliable Computing, T. Csendes, ed., Dordrecht, 1999, Kluwer Academic Publishers, pp. 77–104.
- [32] D. SZOLNOKI, *Algorithms for Reachability Problems*, Shaker, Aachen, 2001.
- [33] —, *Set oriented methods for computing reachable sets and control sets*, Discrete and Continuous Dynamical Systems - Series B, 3 (2003), pp. 361–382.
- [34] W. WALTER, *Gewöhnliche Differentialgleichungen*, Springer-Verlag, Berlin, Heidelberg, New York, Tokyo, 6 ed., 1996.